Hewitt

Upgrading The Solution Series from 5.0 to 5.2

Document Issue: 1.2

Document Issue Status:	First Release
Document Issue Level:	1.2
Document Issue Date:	January 2007
Software Version:	5.2

Contents

Chapter 3   77

Chapter 5   115

P ART 1

Introduction

In This Section

C H A P T E R 1

About This Manual

In This Chapter

Welcome

This manual has been designed to guide you through the process of upgrading your current system (The Solution Series 5.0, 5.01, or 5.02) to the latest version of the system—The Solution Series version 5.2.

Who should use this manual?

This manual is designed to be used by a number of different users. The following users will find it most useful:

- **System administrators**
  If you fulfill the technical role of a system administrator at your company, performing tasks such as scheduling reports and applying program temporary fixes.
- **Project managers**
  If you fulfill the role of the project manager in charge of the upgrade from The Solution Series 5.0 or 5.0.1 or 5.0.2 to the new version 5.2.

Prerequisite skills

Users of this manual should possess a variety of technical skills and authorities, depending on the roles they will play.

System administrator

- Understanding of the job control language or shell scripts for your environment
- Familiarity with job streams
- Understanding of system backup and recovery
- Authority to compile and link production programs
- Understanding of the programs that have been customized
- Programming skills in Cyborg Scripting Language (CSL), Report Generator, and COBOL

Project manager

- Understanding of project management concepts and techniques
- Authority to assign resources

Additional documentation and training

The following documentation is available to help you understand the usage, configuration, and maintenance tasks performed for an implementation of The Solution Series.

Documentation

Document	Description
Using The Solution Series: Administrative Solutions	This prerequisite course documentation covers the introductory concepts and tasks related to The Solution Series. It describes how to navigate through the software and explains the important concepts and functionality of The Solution Series.

Document	Description
Optimizing System Features	This manual provides descriptions of and detailed instructions for performing the configuration and functional administration tasks that support the implementation of The Solution Series.
Technical Administration	This manual provides descriptions of and detailed instructions for performing the technical tasks that support The Solution Series.

If you do not have a copy of any of these documents, you can obtain them from Customer Support.

Training Courses

The following classes are available for customers upgrading from previous versions of eCyborg:

Class	Details
5.2 Webinar	Webinar highlighting the new features and enhancements included in the eCyborg 5.2 release.
eCyborg 5.2 Upgrade Planning	A one-day course during which the upgrade process is discussed.
Upgrading from eCyborg 5.0 to eCyborg 5.2 Workshop	A two-day course during which you begin working on your system upgrade.

If you wish to attend any of these courses, contact Customer Support or visit our website www.hewitt.com/eCyborg for details of course dates and availability.

How this manual is organized

This manual has been organized to make it as easy to use as possible.

	Read this Chapter	To learn about
1	About This Manual	How the manual is organized
		Where to find what you are looking for
		Who should use the manual
		Where to get help
2	What's New in the 5.2 Release	The features and enhancements of eCyborg 5.2
3	Planning your Upgrade	Advice and suggestions to consider when planning to perform the 5.2 upgrades.
4	Implementing The Solution Series 5.2 Release	Detailed steps to get your customizations from the existing production environment into the new 5.2 environment.

	Read this Chapter	To learn about
5	Convert Data	Detailed steps to get your HR data from the existing production environment into the new 5.2 environment.
A	Expand Transactions	Important information on changes to the EXPAND area.
B	Component Lists	Lists of delivered components.
C	Report Generators	Details of delivered RGs.
D	Machine Parameters	Machine parameters
E	Changes to Menu Records	An alternate language version of the menu item title as well as the already existing primary language version.
F	Large Number Changes to Fields	The size of the type 4 data fields changed to accommodate the large numbers enhancement.

Conventions used in this manual

The underlying page layout and design of this manual are meant to be as intuitive as possible for you. Our intent is to make it easy for you to navigate through the manual and concentrate on learning and doing.

Cross-references

Wherever appropriate, we provide cross-references to help you find additional information or further discussion of a specific topic.

☞ *Refer to a cross-reference to find more detail or more discussion on a given topic.*

See also:

■ *A topic to find more detail or more discussion on a given topic (on page no n.)*
For more information.

Notes

Whenever there is important information you should be aware of, we provide a note.

Note: *You will find tips or quick techniques covered in notes.*

Important! or Warning:

Important note or warning.

Identification of platform-specific information

This manual is designed to support The Solution Series on Windows, UNIX, and z/OS operating systems.

Some platform-specific information is offered in tables, according to the format shown here:

Platform	Information
Windows	Windows-specific information
UNIX	UNIX-specific information
z/OS	z/OS-specific information

How to get additional help

If you cannot find the answers to your questions in this manual, contact Customer Support, who will be able to answer specific questions and give you general advice on training.

Please visit our Web site at www.hewitt.com/ecyborg for the latest schedule of available courses and course descriptions.

Suggestions and feedback

We value your feedback on our performance support materials. Please forward any comments on this manual to Customer Support.

CHAPTER 2

What's New in the 5.2 Release

In This Chapter

Overview of the eCyborg 5.2 release

eCyborg version 5.2 includes both new functionality and enhancements to existing functionality. It contains many Payroll, Foundational, Interactive Workforce and both US and Canadian Regulatory enhancements.

Enhancements to Payroll for 5.2

US Garnishments

Enhanced garnishment processing includes the following functionality:

- Calculation of net disposable income
- Automation of multiple child support orders
- Prioritization of multiple garnishment orders
- Automation of Garnishment Rule Maintenance
- Handle the following Garnishment types:
  - Bankruptcy
  - Child Support
  - Child Support Arrears < 12 weeks
  - Child Support Arrears > 12 weeks
  - Spousal Support
  - State/Local Debt
  - Federal HEA & DCA debt
  - Consumer Garnishments
  - Wage Assignments
  - Federal Tax Levy
  - Voluntary Deductions

Note: *The existing 5.1.3 Garnishment process will continue to be available in 5.2 until such time as you activate the 5.2 model on the Organization-To-Rules Cross Reference for Payroll form (AW-SCR).*

A report utility will be provided to list your Child Support data on file. In addition, an Excel spreadsheet will be provided to help map your existing data from the Child Support Garnishment Administration form (PO-SCR) to the new forms. The existing data will then be imported to the new forms with the import wizard.

Differences from 5.1.3

The 5.2 garnishment processing functionality differs from the 5.1.3 process in the following ways:

- **Organization Level**
  The organizational rules will be maintained in tables and calculated automatically by P4CALC, rather than researched and manually applied by the processor as is done with 5.1.3.

  Federal and State protection rule tables delivered with 5.2 are based on the laws known at the time of release. Updates to the rules, thereafter, will be done through Regulatory Bulletins. This will be addressed in the documentation to enhance the maintenance process for the tables going forward.

- **Employee Level**
  In 5.1.3, prioritization of multiple orders for a single employee is currently a manual process involving creating HEDs in a specific order to allow the garnishment with the

highest priority to be deducted first. Prioritization in 5.2 will be automated with report generators and tables.

Accounts Payable

ACH Tape (6A6A) report, Child Support ACH Tape (6I6I) report, and Direct Deposit Register – Child Support (6R6R) have been modified to use the new garnishment L segments built by the GR1SCR – GR4SCR forms. It should be noted that the accounts payable portion of the 5.2 process will function just as the 5.1.3 process does.

New forms

The following new forms have been created for Garnishments:

Org Level	EE Level	Data Delivered	Form	Purpose	Inquiry
X		Yes	Required Deduction HED Types (GRASCR)	Used to establish HED types for deductions required by Federal and state law.	Yes
X		No	HEDs for Required Deduction HED Types (GRBSCR)	Used to establish HED numbers for the HED types required by Federal and state law. *Setup by client required at implementation.*	Yes
X		Yes	Deduction Priority (GRCSCR)	Used to set up the Federal and state deduction priorities for garnishment types, allowing the deductions to be taken in the order required by law.	Yes
X		No	HED Range for Garnishment Type (GRDSCR)	Used to set up HED ranges for each garnishment type. This allows the GR2SCR to determine the next available HED. *Setup by client required at implementation.*	
X		Yes	Federal Tax Levy Protection Rules (GRESCR)	Used to determine the amount of an employee's earnings which is protected by Federal law from a tax levy. These amounts determined by Publication 1494.	
X		Yes	Support Protection Rules (GRFSCR)	Used to determine the amount of earnings which is protected by Federal and state law from a child (or other) support order.	
X		Yes	Garnishment Protection Rules (GRGSCR)	Used to determine the amount of an employee's earnings which is protected by Federal and state law from garnishment types other than Federal tax levies and support orders.	Yes

Org Level	EE Level	Data Delivered	Form	Purpose	Inquiry
X		No	Net Disposable Income Reduction HEDs (GRHSCR)	Used to establish the order in which deductions will be taken to calculate an employee's net disposable income. *Setup by client required at implementation.*	Yes
	X	No	Garnishment Administration 1 of 4 (GR1SCR)	Used to enter a garnishment order for an employee. Completion of this form is required.	
	X	No	Garnishment Administration 2 of 4 (GR2SCR)	Used to enter a garnishment order for an employee. Completion of this form is required.	
	X	No	Garnishment Administration 3 of 4 (GR3SCR)	Used to enter a garnishment order for an employee. Completion of this form is required for tax levies, unnecessary for all other garnishment types.	
	X	No	Garnishment Administration 4 of 4 (GR4SCR)	Used to enter the electronic filing information (ONLY) for a child support order. Completion of this form required for child support orders.	

Modified forms
- **Organization-To-Rules Cross Reference for Payroll screen (AW-SCR)**
  A Garnishments section will be added with these new fields: Garnishment Admin (the control number to tie an organization to a specific set of HEDs); Allow Fees checkbox (to indicate if the client company allows fees to be collected). Entry of a Garnishment Administration control number is required for 5.2 garnishment processing.

Modified option lists
- **PP02**
  New category 'Garnishments 60' added. Used by the A8-SCR to indicate garnishment processing by the new method. When setting up garnishment A8s that will use the new methodology, they must be assigned to this new category.

New option lists
The following option lists have been added for Garnishments:

Option List	Used By
GR01—Garnishment Type	GR1SCR, GR2SCR, GR3SCR, GR4SCR, GRDSCR, GRGSCR
GR02—Garnishment Federal and States	GR1SCR, GR2SCR, GR3SCR, GR4SCR, GRASCR, GRFSCR
GR03—Levy Table Category	GRESCR
GR04—Tax Filing Category	GR1SCR, GR3SCR, GRESCR

Option List	Used By
GR05—HED Types	GRASCR, GRBSCR
GR06—Court/Agency	GR1SCR
GR07—Garnishment Calculation Method	GR2SCR
GR08—Multiple Child Support	GRFSCR
GR09—Deduction Priority	GRFSCR (via radio buttons)
GR10– State Option	GR3SCR
GR12– Garnishment Stop Method	GR2SCR
GR13—Pay Frequency	GRESCR
GR15—Vendor Number Option List	GR1SCR

New report generators

- Calculation of Available Pay—calculates the employee's available pay
- Net Disposable Pay—calculates the employee's net disposable income based on the order type and issuing state
- Federal Tax Levy Protected Pay—calculates an employee's protected pay when a federal tax levy is processed
- Calculate Support Assignments—calculates an employee's protected pay when a support order is processed
- Other Garnishments—calculates an employee's protected pay when any other order type is processed
- State specific rules governing the calculation of protected earnings are entered on the Support Protection Rules form (GRFSCR) and Garnishment Protection Rules form (GRGSCR) and are delivered with the system.

Tax reciprocity enhancements

When reciprocal taxation is active for an organization (by way of the "Reciprocal Rules" option on the Organization Options form (AF-SCR)), an employee's reciprocal code is now determined automatically during the pay run. To accomplish this, the processing program, P4CALC, has been modified to recognize the Resident/Work State value (Option List PR10) on the Employee Tax Record Maintenance form (JJ-SCR). The following end-user changes are a result of this enhancement:

- The Reciprocity Rule on the Employee Tax Record Maintenance form (JJ-SCR) is read-only, and is updated following a pay run if necessary.
- Time entry overrides to the employee's normal work location will invoke the automatic reciprocity rule look-up.
- The Reciprocal Tax Mass Maintenance form (JRMSCR) has been removed from the system.
- A new pay run report, "Reciprocal Rules Report, Rules in Effect by Employee" (5R5R), has been created. This report lists the employees who have different resident and work locations and displays the associated reciprocal rule and tax reduction if appropriate.
- It is no longer necessary to use the State Reciprocal Tax Setup form (JR-SCR) to enter an employee's reciprocal code. This form is still available for informational purposes.

Tax Arrears

The ability to place uncollected taxes into arrears for an employee has been added to the payroll system. This situation can occur when an employee is paid large amounts of imputed income. Modifications to the system for this enhancement include:

- Addition of the Tax Arrears switch on the Organization Options (AF-SCR) screen to activate this feature.
- Changes to the Payroll Audit Trail warning the user that a tax was not withheld.
- Addition of the Tax Arrears Report Generator (3U3U) which will produce a report of employees that taxes go into arrears during the current tax run.
- Addition of the Tax Arrears Balance (JTASCR) screen which will display an employee's current tax arrears and balances and allow for a manual update of the arrears and arrear status.

Reformatted Payroll Reports

The 5.2 release incorporates enhancements to the Combined Register and adds two new payroll reports.

Combined Register (2222)

Changes have been made to improve the usability of the Combined Register. At a high level, the changes/updates are as follows:

- The Combined Register report can now be sorted by Controls 3, 4, 5, and 6 by using a new multi-paneled form, the Register Overrides form (CK-SCR and CK2SCR).
- An employee name consisting of up to 30 characters will display.
- The 2T2T Tax Filing report is being renamed "Tax Register".
- The Combined Register now includes the report Memo HED Register (2M2M and 2M2M Combreg) as an optional report.

Remittance Summary (2L2L)

This new report provides a summary of deductions to enable easier identification of remittances, such as tax payments and remittances to other government agencies.

The parameters can be defined to have a summary of selected HEDs and Tax Bodies, and the report produces a total across organizations having the common setup parameters. The report is independent of the CONTROL-COUNTRY to which an organization is set up.

Time Entry and Analysis

There are 10 new reports run from the Report Group Activities form (RGMSTR). The reports can be run at any time during a payroll run to enable you to review and approve the time entries and adjustments entered into the system (or identify any that require revisions).

The 10 new reports (PR71PT, PR72PT, PR73PT, PR74PT, PR75PT, PR76PT, PR77PT, PR78PT, PR79PT, and PR7APT) each display the relevant information available on the Employee Database (FILE02). The following table identifies the report and the screen records it displays:

CSL Report	Display records from:
PR71PT	TC1SCR, TC1EDT, TCFSCR, TCFEDT
PR72PT	TC2SCR, TC2EDT
PR73PT	TCBSCR, TCBEDT
PR74PT	KA-SCR
PR75PT	KB-SCR
PR76PT	KC record entered in KA-SCR. KB-SCR
PR77PT	KD-SCR, KG-SCR
PR78PT	KF-SCR, KH-SCR
PR79PT	KL-SCR
PR7APT	TCMSCR, TCMEDT

The 10 reports are designed to be run together from a user-created report group (RG) from the Report Group Activities form (RGMSTR) but can be run separately as individual elements. Additionally, any configuration of the reports can be grouped into a report group.

Retroactive Pay Processing

This is an automatic process to calculate back-dated pay changes. This new feature will automatically process retroactive pay calculations for salary records entered on the Salary Assignment/Changes form (40-SCR) with an effective date earlier than the start date of the current pay period for the employee.This process will now support FLSA premiums on payments generated within the 5.2 system.

New forms

The following forms have been added for Retroactive Pay Processing:

Form	Description
Retroactive Pay Process Parameter form (RPPSCR)	This form is used to store the Retroactive Pay Process Parameter details. It will allow a retroactive payment of Non Overtime/Overtime HEDs to be routed to another HED element and/or cost center(s).
Retroactive Pay Method Rules form (RPMSCR)	This form is used to store the Retroactive Pay Method Rules details. It will allow a retroactive payment of a particular HED to be routed to another HED element and/or cost center(s).

Retroactive Pay Control Settings form (RPCSCR)	This form is used to store the Retroactive Pay Control Settings details. It controls if retroactive pay calculations are required automatically.
Payroll Extended/Misc. Details form (PEMSCR)	This form will be used as a multipurpose employee payroll form. For retro pay, the form will be used to hold employee-level Retroactive Pay Rule details.

New report
- **Automated Retroactive Processor report (ARPRPT)**
  Parameters form: A-ARPR
  This report is used to read the QRT (Retroactive Trigger) records and create retroactive payments based on rules as defined on the supporting Retroactive forms.

Modified form
- **Organization-To-Rules Cross-Reference For Payroll form (AW-SCR)**
  Amended to add 'switch' to turn on automated retroactive pay functionality.

Foundational Enhancements for 5.2

Security Enhancements

Significant enhancements have been made to eCyborg security to comply with the tightened security requirements imposed by the Sarbanes Oxley Act. Security Officers wishing to know more about these security features should refer to the Setting Up and Maintaining Security guide or contact their Hewitt customer support representative for more information.

Extended security features include:

- Password encryption using SHA-1
- Minimum password size for both Admin users and other users
- Minimum number of alpha characters in passwords
- Minimum number of numeric characters in passwords
- Mandatory password change after a configurable number of days
- No password reuse within a configurable number of days
- No password reuse using a configurable number of previous passwords
- Password expiration date
- Set up of initial password
- Account locking after configurable number of inactive days
- Account removal after configurable number of inactive days
- Account locking after configurable number of unsuccessful logon attempts
- Support for multiple security officers (One-on-one security)

Three new forms have been created for Security Officers to maintain password rules and to manage user accounts:

- Extended Security Setup (SECEXT)
- User Account Setup (SECUSR)
- Inactive Accounts - Removal (SECRMV)

Important! The security enhancements do **not** support Flexible security. It is recommended that you configure your security so that only One-on-one security is used before enabling this functionality. Refer to the Setting Up and Maintaining Security manual for more information on Flexible and One-on-one security.

HTML-Enabled Content

This requirement impacts how HTML resource strings are filtered to remove potentially unsafe tags.

The HTML resource strings will be filtered before being used to remove all tags that are not in a predefined list of acceptable tags. In order to allow you to modify resource strings using other tags, an alternative list of allowed tags can be added to the IW database in the IW_CONFIG table, by adding a property named "resourceTags", whose value is a comma and/or space separated list of tag names.

Multiple activities on the same date

Modifications have been to the Major Activities segment (LZC) to allow for multiple activities to be effective on the same date (such as a layoff and a rehire). The main part of this modification is the addition of key separator in the key of the Major Activities (LZC) segment. The key separator ZC-KEY-SEPARATOR has been placed in between the ACTIVITY-CODE and the ACTIVITY-DATE fields.

Changes to EXPAND records

We have significantly enlarged the sizes of the Company (PAYER), Employee (PAYEE), and Tax area.

P4CALC/O4CALC

In P4CALC/O4CALC, the PAYER and PAYEE areas have been increased by 100,000 characters each. Also, the TAX AREA has increased:

EXPAND Area	New area sizes for 5.2 (bytes)
Company area (PAYER)	132,271
Employee area (PAYEE)	124,788
TAX AREA	784,000

Note: *You have the option to reduce the new size of the TAX AREA. If a "T" is added to the Machine Parameter string, then 720,000 bytes will be removed leaving 64,000.*

CBSV

In CBSV, AREA 2 and AREA 4 have been increased by 100,000 characters each:

EXPAND Area	New area sizes for 5.2 (bytes)
AREA 2 (employee)	129,158
AREA 3	232,767
AREA 4 (company)	132,293

Important!

The pre-5.2 EXPANDs will not function in extracting programs from the 5.2 CYBMST. You will need to remove or modify your EXPAND control records before upgrading to 5.2.

Enhancements to Interactive Workforce for 5.2

JDBC driver to replace STAPI in Interactive Workforce architecture

A new JDBC driver is now used by Interactive Workforce (IW) to communicate with eCyborg instead of the previous API (STAPI). This removes the need for Orbix.

Interactive Workforce JSP request parameters

An enhancement has been made so that each IW JSP is checked to ensure employee key information is not being passed as a request parameter. Any JSPs that are found to be passing key information as a request parameter will be modified. This requirement removes the request parameters used in JSPs and instead codes them as setters and getters in Java bean. This makes the employee data secure from unauthorized access.

Option to remove national ID/SIN/SSN from all Self Service Screens

To comply with privacy laws an option is now available in Interactive Workforce (IW) to prevent IW from displaying the ID/SIN/SSN of the employee's family members and beneficiaries. The employee will still be able to enter it, but once entered either by the employee or directly into The Solution Series, it will not be displayed again in IW.

Interactive Workforce will also no longer display the IS/SIN/SSN on the employee's pay advice page.

Note: *In the UK, Ireland, and Jersey, this is National Insurance Number, PPS Number, and Social Security Number.*

Security enhancements

Many security enhancements have been added to ensure that Interactive Workforce complies with the Sarbanes Oxley Act.

Data encryption

It is now possible to have the initial password column in the Interactive Workforce Cyborg Users database table display as hashes (#) and encrypt a password distribution file containing a notification email address.

Enhanced interface

Interactive Workforce has been given a new look, consistent with the Hewitt family of products. The following are examples of the new interface:
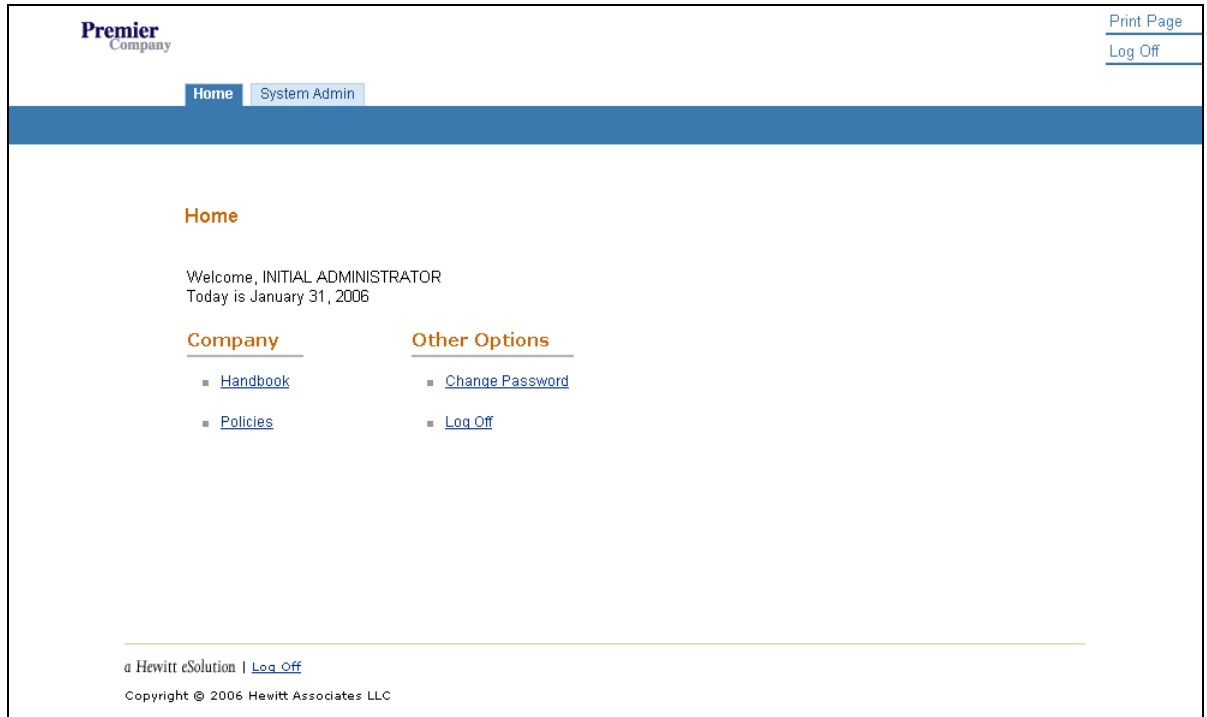
Print Page

Log Off

Home   System Admin

System Admin

Interactive Workforce

- User Account and Password
- Account and Password Policies
- Generate New User Passwords
- Features Management
- System Maintenance
- Resulting Statuses Selection
- Organization Selection
- Option List Refresh

Benefits

- Benefit Plan Groups
- Benefit Plan Subgroups
- Benefit Plan Display Order
- Select Flex Master Plan
- Flex Master Plan Maintenance
- Benefit Plan Maintenance
- Dependent Coverage
- Beneficiary Numbering

Human Resources

- Dependent Numbering
- Requisition Descriptions

Payroll

- Direct Deposit HED Set-Up

Problem Notifications, Program Temporary Fixes, and Defect fixes in 5.2

Fixes resulting from customer feedback have been integrated into the system. Feedback has come via Problem Notifications from WebFirst, Program Temporary Fixes posted on CUBBS, and defects identified by the Quality Assurance group.

PTF and PN fixes have been packaged together in the 5.1.1, 5.1.2, and 5.1.3 service packs. These service packs have been rolled into the product, and additional PTF and PN fixes have been rolled in.

☞ *For details on the PNs and PTFs included in the 5.2 release, refer to the Listing of PTFs and PNs included in the 5.2 release excel spreadsheet, available for download from the 5.x Documentation area on the Customer Center.*

US Regulatory Enhancements included in 5.2

The following regulatory bulletins are included in the 5.2 release. Please refer to the relevant RB documentation for more information on any of the RBs.

BERKS EIT - PA Locals (RB04-005)

This bulletin provides the necessary tax-related updates to meet the Berks E. I. T. Bureau requirements for Pennsylvania locals with 50 or more employees. Berks collects information on the employees' earned income tax and submits Quarterly Form 511.

California Meal Penalty and Puerto Rico Overtime (RB04-007)

This Regulatory Bulletin brings the Time and Attendance application in compliance with the following California and Puerto Rico laws:

California Meal Periods

Employees are entitled to a minimum of a thirty (30) minute duty-free meal period for every five (5) hours worked. A second meal period is required if an employee works more than ten (10) hours per day. Meal periods may be waivered by mutual consent. Employees, whose total work period is no more than six (6) hours, may agree to waive the meal period requirement providing their employer concurs. (Labor Code § 512) If an employer fails to provide a meal period, the employer must pay one (1) additional hour of pay at the employee's regular rate of pay.

Puerto Rico Extra Hours of Work

Except when fixed by the Minimum Wage Board or by a collective bargaining agreement, or both, an employer must pay double the regular rate for extra hours of work. Employers covered under the federal FLSA must pay 1 ½ times the employee's regular rate for hours worked over 8 hours a day.

Puerto Rico defines extra hours of work as follows:

- Over eight hours in any period of 24 consecutive hours
- Over 40 hours during any week unless the hours worked daily over eight are paid at double the regular rate
- In the day when the establishment is closed to the public by law unless the employer secured a permit from the secretary of labor and the total hours worked during the day is not over eight hours or 40 hours during the week
- On days of rest for business and industries not subject to the closing of their establishment
- Over the maximum hours set by the Minimum Wage Board for the occupation, business, or industry in question
- Over the maximum hours a day fixed in a collective labor agreement

Puerto Rico also requires that employees receive a meal time after the fifth hour of consecutive work.

Tax Authority File Enhancement (RB04-013)

The current tax code structure for local taxing authorities maintained on the TAXFILE may combine municipality and school district rates into a single tax code. With the release of RB04-013, all combined municipality and school district rates are separated into distinct tax codes. Additionally, the TAXFILE has been enhanced to include all tax codes. The following is a list of the changes:

- Updates to the TAXFILE
- TAXFILE loading enhancements
- Conversion to add municipal and school district tax authorities

Updates to the TAXFILE

The following updates were made to the TAXFILE:

- Local tax authorities formally maintained in the Maintaining Payroll Tax Codes manual are now maintained on the TAXFILE.
- Created new tax codes and rates for each school district in Pennsylvania.

TAXFILE loading enhancements

With over 3,000 records being added to the TAXFILE in this one release, the audit trail will increase by several hundred pages. In response, new functionality has been added to P4CALC to perform the following:

- All tax authorities are now loaded to the master file without placing a 'A' into the LOAD TAX BODY field of the T1 transaction. To accomplish this step, you now have the option of placing a 'A' in position 27 of the H2 parameter transaction (P05RDR file for P4CALC step).

- P4CALC has been modified so that only company tax fields which are actually changing are printed in the IS/WAS section of the Payroll Audit Trail. To force the IS/WAS to print all company tax fields, even those which are not changing, you can specify a value of 'A' in position 28 of the H2 parameter transaction (P05RDR file for P4CALC step).

Conversion to add municipal and school district tax authorities

A new one-time Report Generator (9L9L) and WL transactions were included in this bulletin. The Report Generator adds Employee 'J' batch transactions for employees. The transactions add the Pennsylvania school district tax record for employees who have a municipal tax code for which a school district tax code was created. Once the Report Generator is run, it is not copied in the output P20 Payroll Master File.

Unemployment/Disability Insurance Calculation Enhancements (RB04-017)

This bulletin provided the following modifications:

System fields that have been modified

The following fields have been redefined to more clearly report wages and taxes:

- FLD 121, which previously reported YTD uncutoff (total) UI wages, now reports YTD cutoff (taxable) UI wages
- FLD 123, which previously reported current, MTD, and YTD OT Premium, now reports uncutoff (gross) current, MTD, QTD, and YTD UI wages
- FLD 118, which previously reported UI Total Wages for state tax authorities, now reports current, MTD, QTD, and YTD Resident Wages
- OT Premium amounts have been removed from the employee's tax segment and are now stored in a new employee segment
- Fields 117 through 124 now display for all tax types (with the exception of 101, 103, 104, and 105) on the following forms:
  - Taxes To-Date Inquiry (JT-SCR)
  - Tax Adjustment (KD-SCR)
  - Tax Adjustment - Alternate (KG-SCR)
  - Employee Profile (PROFIL)

Overtime premium storage modifications

Overtime premium amounts, previously accumulated in Field 123 and stored on the employee's tax segment, are now accumulated and stored on a new employee segment (L7D). A new Overtime Premium form (OP-SCR) allows you to override an incorrect overtime premium amount for an employee.

Existing forms that have been modified

The following forms have been modified:

- **Organization Options (AF-SCR)**
  - Frick Tape field has been removed
  - A new field for unemployment insurance wages accumulation, (UI Accum) has been added. This gives you the option to decide, at an organization level, whether unemployment (and disability, where applicable) insurance wages will be accumulated in the permanent work state or in the override work state when a time entry location override occurs.

- **Taxes To-Date Inquiry (JT-SCR)**
  - Tax code 102 Federal Withholding tax:
    Field 121—field label Federal UI Wages is now Taxable FUI Wgs
    Field 123—field label OT Premium is now Gross FUI Wages

  - Tax codes beginning with 2:
    Field 118—field label UI Total Wages is now Resident Wages
    Field 121—field label State UI Wages is now Taxable SUI Wgs
    Field 123—field label OT Premium is now Gross SUI Wages

  - Tax codes beginning with city or county:
    Field 121—field label State UI Wages is now Taxable UI Wgs
    Field 123—field label OT Premium is now Gross UI Wages

  - Tax codes beginning with 4, containing a DI in positions 5 and 6 of the Tax ID:
    Field 121—field label Disability is now Taxable DI Wgs
    Field 123—field label OT Premium is now Gross DI Wages

- **Tax Adjustment (KD-SCR)**
    - Field 121—field label Unemploy Wages is now Taxable UI Wgs
    - Field 123—field label Premium is now Gross UI Wgs

- **Tax Adjustment - Alternate (KG-SCR)**
    - Field 121—field label Unemploy Wages is now Taxable UI Wgs
    - Field 123—field label Premium is now Gross UI Wgs

- **Employee Profile (PROFIL)**
    - Tax code 102 Federal Withholding Tax:
    Field 121—field label Fed UI Wages is now Taxable FUI Wgs
    Field 123—field label OT Premium is now Gross FUI Wages

    - Tax codes beginning with 2:
    Field 121—field label State UI Wages is now Taxable SUI Wgs
    Field 123—field label OT Premium is now Gross SUI Wages

    - Tax codes beginning with 3:
    Field 121—field label UI Wages is now Taxable UI Wgs
    Field 123—field label OT Premium is now Gross UI Wages

    - Tax codes beginning with 4, containing a D in position 54 of the T3-transaction of the TAF:
    Field 121—field label Disability Wages is now Taxable UI Wgs
    Field 123—field label OT Premium is now Gross UI Wages

Existing report generators that have been modified

The following report generators have been modified:

- Active Employees on the 12th of the month (2B2B)
This report, which previously used field 121 to report gross UI wages for the quarter, now uses Field 123.

- Control Headers (0103)
Frick Tape flag has been removed from the Organization Options form (AF-SCR). In its place, unemployment insurance accumulation (UI Accum) data is reported. The new column label is UI ACCUM.

- Master File (0202)
The following headings have been changed:
    - UNEMP is now Taxable UI Wgs
    - O.T. Premium is now Gross UI Wgs

- Tax Filing Report - All Frequencies (9091)
Overtime premium amounts are no longer reported here. The following headings have been changed:

- ▪ FUI Wages is now Taxable UI Wages
- ▪ SUI Wages is now Taxable UI Wages
- ▪ Disability Wages is now Taxable DI Wages
- ▪ Premium O.T. is now Gross UI Wages or Gross DI Wages
- ▪ Premium O.T./QTD-SUI TTL is now Gross UI Wages

- ▪ Tax Register Report - Paid Frequencies Only (2T2T)
  Overtime premium amounts are no longer reported here. The following headings have been changed:
  - ▪ FUI Wages is now Taxable UI Wages
  - ▪ SUI Wages is now Taxable UI Wages
  - ▪ Disability Wages is now Taxable DI Wages
  - ▪ Premium O.T. is now Gross UI Wages
  - ▪ Premium O.T./QTD-SUI TTL is now Gross UI Wages

- ▪ Tax Filing Report - Paid Frequencies Only (9090)
  This report does not feed into the Combined Register Report (2222). Overtime premium amounts are no longer reported here. The following headings have been changed:
  - ▪ FUI Wages is now Taxable UI Wages
  - ▪ SUI Wages is now Taxable UI Wages
  - ▪ Disability Wages is now Taxable DI Wages
  - ▪ Premium O.T. is now Gross UI Wages
  - ▪ Premium O.T./QTD-SUI TTL is now Gross UI Wages

- ▪ Workers Compensation (7W7W)
  This report, which previously obtained overtime premium amounts from an employee's tax segment, now obtains the amounts from a new employee segment (L7D).

2005 Reporting Distributions to 1099-R Recipients (RB04-021)

This bulletin provides the necessary instructions to take advantage of the Combined Federal/State Filing (CF\SF) Reporting for 1099-R recipients at year-end 2005.

To take advantage of this program, you must report 1099-R distributions by type of distribution. Following is a list of the 2004 1099-R distribution types and their codes.

Code	Explanation
1	Early Distribution, no known exception
2	Early (premature) distribution - exception applies other than disability or death
3	Disability
4	Death (includes payments to a beneficiary)
5	Prohibited Transaction
6	Section 1035 Exchange
7	Normal Distribution
8	Excess Contributions plus earnings/excess deferrals and/or earnings taxable in the current year

9	PS 58 Costs
A	Qualifies for ten-year averaging (Code A) and can be specified only when the Box 7 Primary distribution code is numeric
D	Excess contributions plus earnings/excess deferrals taxable two years ago
E	Excess Annual Additions under section 415
F	Charitable Gift-Annuity
G	Direct rollover & Rollover Contribution to a Qualified Plan
J	Early distribution from a Roth IRA, no known exception
L	Loans treated as deemed distributions
R	Recharacterized IRA contribution made for current year
P	Excess contributions plus earnings/excess deferrals taxable last year
Q	Qualified Distribution from a Roth IRA
R	Recharacterized IRA contribution made for the previous year
S	Early distribution from a SIMPLE IRA in the first two years, no known exception
T	Roth IRA distribution, exception applies

Use one of the following three methods to report 1099-R distributions and generate the necessary magnetic media with K records using the Hewitt year-end processor for year-end 2005 federal/state filing:

▪ Set up a separate 1099-R company for each distribution type used by your organization.
OR

▪ In a 1099-R company, set up a separate employee record for each type of distribution the employee receives.
OR

▪ Use the new procedure that follows to include all recipients of 1099-R distributions in a separate 1099-R company and set up HEDs and taxes for 1099-R reporting.

Important: For all of the above methods, 1099-R recipients must be in a retirement organization (control 1-2), and separate checks must be issued for distributions sent to different states. For example, if the recipient moved between states during a pay period, two checks must be issued, one for each state.

Supplemental Wages over 1,000,000 (RB04-023)

The American Jobs Creation Act of 2004 provides for withholding at the highest incremental rate for supplemental wages in excess of $1,000,000. For 2005 this rate is 35%. The 25% withholding rate continues to apply to supplemental wages less than $1,000,000.

Supplemental wages may include bonuses, commissions, payments for accumulated sick leave, severance pay, awards, back pay and retroactive pay increases, and payments for non-deductible moving expenses. This withholding rate requires organizations to total the supplemental wages for an individual employee across companies when determining the

withholding rate. A new report, SSNs with Supplemental Wages, lists employees who received cumulative supplemental wages of $1 million or more year-to-date.

Hewitt has developed a process that will allow you to more easily track supplemental wages paid to employees by a control group of companies as defined by the Internal Revenue Service.

Emergency and Municipal Services Tax (RB05-005)

Act 222 of 2004 amends the Local Tax Enabling Act, Act 511 of 1965, to permit municipalities and school districts (except the Pittsburgh School District)1 to impose a combined Emergency and Municipal Services Tax (EMS tax) of up to $52 a year beginning on and after January 1, 2005. The EMS tax replaces the occupational privilege tax.

Supplemental Tax Phase II (RB05-016)

This bulletin provides the necessary updates to meet the taxation on supplemental wages set forth in the American Jobs Creation Act of 2004.

Currently, in Solution Series employer taxes are setup as deductions (Memo HEDs). With this bulletin, an employer TAXFILE contains new or updated employer/employee tax records. There are (4) types of tax categories:

- Hours Based
- Wage Base (x) Tax Rate
- Employee Count (x) Fixed Dollar Amount
- EMST for Pennsylvania

Remove Social Security Numbers from Pay Advices (RB05-025)

California passed a law governing the use of social security numbers that is summarized by the Office of Privacy Protection, Department of Consumer Affairs. Effective January 1, 2008, only the last four digits of a Social Security Number or existing employee identification number other than a Social Security Number, may be shown on a payroll check.

To protect employee privacy, Hewitt is releasing the overrides and updates contained in this bulletin now to remove the Social Security Number (SSN) and Social Insurance Number (SIN) from all external employee informational reporting not being used for tax filing purposes.

The documents targeted for modification are the Payroll Check and Payroll Deposit Advice in addition to various payroll reports. Internal reports and reports used for tax filing and reporting purposes will remain unchanged. The report change removes all instances of the SSN and SIN and displays only the employee number (10-digits). The modifications made with this project affect the eCyborg Payroll Module.

Important! California law requires that employers remove their employee's social security number from payroll reporting information no later that January 1, 2008. For your employee's privacy, however, you may want to do it sooner.

US Tax regulatory bulletins included in 5.2

All tax bulletins from RB04-001 up through the RB06-003 tax regulatory bulletin, have been incorporated into the 5.2 release:

- Tax Bulletin (RB04-001)
- Tax Bulletin (RB04-002)
- Tax Update (RB04-010)
- Tax Update (RB04-011)
- Tax Update (RB04-012)
- Tax Update (RB04-015)
- Tax Update (RB04-022)
- Tax Update (RB04-025)
- Tax Update (RB05-001)
- Tax Update (RB05-002)
- Tax Update (RB05-003)
- Tax Update (RB05-004)
- Tax Update (RB05-006)
- Tax Update (RB05-012)
- Tax Update (RB05-014)
- Tax Bulletin (RB05-017)
- Tax Bulletin (RB05-019)
- Tax Bulletin (RB05-020)
- Tax Bulletin (RB05-021)
- Tax Bulletin (RB05-023)
- Tax Update (RB05-024)
- Tax Bulletin (RB05-026)
- Tax Bulletin (RB05-029)
- Tax Bulletin (RB05-033)
- Tax Update (RB06-001)
- Tax Update (RB06-002)
- Tax Update (RB06-003)

Please refer to the relevant RB documents for further details on the contents of each tax bulletin.

All subsequent tax regulatory bulletins must be applied to the 5.2 release.

US Quarterly regulatory bulletins included in 5.2

The latest available Quarterly regulatory bulletins (Quarter 1, 2004 through Quarter 4, 2005) has been included in the 5.2 release.

- 2004 Q1 Quarterly Processor (RB04-004)
- 2004 Q2 Quarterly Processor (RB04-008)
- 2004 Q3 Quarterly Processor (RB04-014)
- 2004 Q4 Quarterly Processor (RB04-019)
- 2005 Q1 Quarterly Processor/Electronic Filing (RB04-004)
- 2005 Q1 Berks Quarterly (RB04-008)
- 2005 Q2 Quarterly Processor (RB05-015)
- 2005 Q3 Quarterly Processor (RB05-022)
- 2005 Q4 Quarterly Processor (RB05-030)

Please refer to the relevant RB documentation for further information on any particular bulletin.

All subsequent Quarterly bulletins must be applied to the 5.2 release.

Canadian Regulatory Enhancements included in 5.2

The following Canadian regulatory bulletins are included in the 5.2 release. Please refer to the relevant RB documentation for more information on any of the RBs.

Quebec Provincial Taxability Code (RB05-010)

Provincial Changes

Québec

Québec requires special provincial taxation rules around non-benefit earnings (stock payout for Québec employees). Québec requires part of the payment to be provincially taxable and not federally taxable.

A new taxability code (Option List PP09) value 21 – QC Prov/QPP/EI only was created to withhold provincial tax only.

Overrides to P4CALC are required to accomplish this taxability change.

Canadian tax updates included in 5.2

The following Canadian tax bulletins have been incorporated into the 5.2 release:

- Tax Bulletin (RB05-018)
- QPIP Tax Update (RB05-028)
- Tax Bulletin (RB05-032)
- Tax Update (RB06-004) (ROE Web)

Please refer to the relevant RB documents for further details on the contents of each tax bulletin.

All subsequent tax regulatory bulletins must be applied to the 5.2 release.

Overview of the eCyborg 5.2 Release (5.0 to 5.1)

The 5.2 release of The Solution Series and eCyborg offers a wealth of new features and functionality. Many enhancements are delivered as part of the core Solution Series product, where other exciting features are provided within the eCyborg product, including enhancements to Interactive Workforce, the Web Client, Analytics, and the Collaborative Platform.

This section provides an overview of the changes that occurred between versions 5.0 and 5.1.

Note: *Support has changed for operating systems and third party products with this release. Please refer to the Customer Support Center at www.hewitt.com/ecyborg for technical prerequisite details.*

- Enhancements to The Solution Series core system
  - Relational Database enhancements
  - Support for bilingual French and English
  - Support for communication events triggered from the web client, Interactive Workforce, or the Collaborative Platform
  - Large number support for amounts larger than 9,999,999.99
  - Automatic employee number assignment
  - Enhanced setup for the Benefits Recalc Report (85-RPT)
  - New Workforce Planning functionality
  - 24/7 access for online updates during payroll runs

- Enhancements to Reporting Administration
  - Data mart and Catalog enhancements--additional employee data and applicant tracking data have been added to the delivered data mart and catalogs
  - Upgrade macros and methodology that will simplify the work required to keep the catalogs up to date
  - Support for the incremental extraction of employee data to the WorkForce Data Mart
  - Support for Web Impromptu (requires Professional Services assistance)
  - Bilingual catalogs available with Canadian bilingual product

- Enhancements to Interactive Workforce
  - New workflows for employee name and address changes, employee competency updates, and job applications
  - Support for mid-year Benefit enrollments for life event changes
  - Support for employee benefit rollover selections from one year to the next
  - Support for single sign-on functionality from a company portal

- Web client enhancements
  - Ability to configure environments using the web client

- Analytics enhancements
  - Four new PowerPlay models:
    - Turnover Analysis
    - Time Away (Absenteeism) Analysis
    - Work Related Injury Analysis
    - Compensation Change Analysis
    - Support for the Cognos Upfront portal (requires Professional Services assistance)

- Enhancements to the Collaborative Platform
  - Ability to write data updates to The Solution Series from outside applications
  - New strategic partners (requires Professional Services assistance)

- Problem Notifications, Program Temporary Fixes, and Defect fixes
  - Fixes resulting from customer feedback have been integrated into the system. Feedback has come via Problem Notifications from WebFirst, Program Temporary Fixes posted on CUBBS, and defects identified by our Quality Assurance group. The PTF and PN fixes are in addition to those accumulated and delivered in the 5.0.1 and 5.0.2 service packs.

- North American Regulatory enhancements
  - Citizenship Country code (HR05) option list (RB02-030)
  - US Electronic funds transfer of child support (RB02-018)
  - US INS compliance (RB02-032)
  - US Equal Employment Opportunity (EEO) and Federal Contractor Veterans' Employment (VETS-100) reporting updates (RB02-022 and RB02-028)
  - US HIPAA Electronic Data Transfer enhancements (RB02-033)
  - US HIPAA Privacy enhancements (RB03-004)
  - US Supplemental Wage enhancements (RB02-024)
  - US California Flat Tax Rates (RB03-001)
  - US Tax regulatory bulletin updates (through RB03-010)
  - US Quarterly regulatory bulletins (through Q1 2003--RB03-007)
  - Canadian Tax updates (RB02-036 and RB02-038)

Enhancements to The Solution Series core system between 5.0 and 5.1

Relational Database enhancements

With the 5.2 release of the system, we introduce the enhanced relational database system.

The new RDBMS provides:

- Improved Commit, Rollback and Recovery
- Data keys for Company/Organization, Employee and Tax removed from FILE02
- Improved error handling
- New CSL requirements
- Changes to the Installation Process
- Changes to the process for applying PN fixes

Technical Considerations

Improved Commit, Rollback, and Recovery

With our intention to improve upon several key points of relational processing, we believe the improvement of our commit/rollback methodology provides the foundation on which to build. Our commit/rollback methodology enforces a logical unit of work through out CBSVO/B for update, add and/or delete for both on-line and background transactions on the System Control Repository (FILE01) and the database. This includes program O4CALC, a member of CYBMST.

The Relational Database Management System (RDBMS) is the primary repository of application data. Database backups are an important part of a comprehensive database protection strategy and are the responsibility of the customer. Our enhancements ensure the 100% synchronization of the System Control Repository (FILE01) and the database.

For the System Control Repository (FILE01), we now maintain a copy on the database in TABLE01. The exception to this is 'ZL' (lock) records. For FILE02, we have moved the company, employee, tax and various Z records to two new tables on the database. Table ZZ2IND contains the Company/Organization, Employee and Tax keys. The various Z (other) records will now be stored in table, Z_TABLE. The exceptions to this are the temporary records, ZN records, scratch area records created by the CSL verb SCRATCH; ZT records, scratch area records created by the CSL verb SCRATCH; ZQ records, records used to pass data back and forth with O4CALC during PAY-CP; ZX records, executable code which is refreshed from System Control Repository (FILE01) if not found on FILE02.

Record	Description
ZE	Application Error record
ZH	These records are created by Training Administration, Position Management, and WRITER when necessary to keep track of where a user currently is in a process
ZI	IS/WAS Audit record, used in producing the IS/WAS Audit Report
ZR	Report Viewing records, are created when the VIEW feature is used

Record	Description
ZU	Batch Balancing records, are created as a result of batch balancing time entries
ZV	Batch Time entry records
ZY	Session records
ZZ	Audit records
ZZA	Time Entry records

RDBMS vendors ensure that if any errors occur during a transaction, the database uses the information in the rollback log file to roll back the transaction. In our rollback strategy, we include the database changes made but not committed and the corresponding System Control Repository (FILE01) updates. We store System Control Repository (FILE01) keys that have been involved in adds, changes, and deletions in a working storage array, TABLE01-TABLE, prior to the logical unit of work (LUW) being committed.

For background (batch) transactions and high volume I-O to the System Control Repository (FILE01), we have created a system configurable commit limit within CBSV. SQLLIM, configurable via the EXPAND screen, is the minimum # of SQL transactions before COMMIT. Generally, the SQLLIM is set high to improve performance. SQLLIM is initialized to 5000 in CBSVO/CBSVB. During batch DEMO and PAYMRG 171 processing, the limit also set to 5000. The value of SQLLIM is dependent on the amount and size of the rollback logs available for use. If the limit is reached prior to the completion of background (batch) process, we will force a commit to maintain the integrity of the rollback files.

Data Keys for Organization/Company, Employee, Tax keys removed from FILE02

The organization/company, employee, and tax keys are no longer resident on the Employee Database (FILE02). With this improvement, all organization/company, employee, tax and various Z record data exist on the database where data synchronization would now be taken care of by the RDBMS vendor. The Employee Database (FILE02) now contains Batch Payroll Master (Report Generators), CSL executable programs, temporary Z records, SUBMIT/VIEW report output and error log records.

Improved Error Handling

A new Error Logging and Display facility (E.L.D.) logs and communicates errors from SQL, COBOL, and other 3rd party tools. Once the ZE errors are logged to FILE02, the E.L.D. facility performs a rollback if necessary, and either terminates, or returns to the paragraph from which it was invoked. The E.L.D. facility is utilized for relational and non-relational systems, since not all errors are relational in origin. This solution provides for error severity level validation, which allows users to continue in the event of a non-fatal error. There are two categories of errors: Reject or Terminal. Rejects are those errors that are informational and allow the user the ability to continue working (for example, invalid password). Terminal errors indicate a failure during processing, and ultimately abort the process or transaction. These errors are recorded on FILE02 for evaluation. A new utility, DSPERR, will print formatted error information. The ZE errors on FILE02 will be cleared during a PAYMRG 171 process.

Execute the DSPERR utility in the background as follows:

INPUT	FILE01	System Control Repository
	FILE02	Employee Database
	FILE04	Control Record File
OUTPUT	FILE03	Error Log File
EXECUTE	CBSVB	

The control record on FILE04 has the following syntax:

In these positions	Enter	Description
23–28	DSPERR	program name

The layout of the ZE record is as follows:

```
  Date> aaaaaaaa    Time> bbbbbbbb  ccc Session> dddd  Break> ee

  Organization: ffffff             Last FILE01 Key: ggggg
   Record Type: h                          Program: iiiiii
   Employee No: jjjjjjjjjj             File Status: kk
     Master No: llll                 Cyborg Error: mmmmmm
    Segment ID: n  oooooooooooo          Paragraph: pppppppppppppppppppppppp
       User ID: qqqqqqqqqqqqqqqqq        SQL Error: rrrrr

 Error Message: sssssssssssssssssssssssssssssssttttttttttttttttttttttttttttt
                uuuuuuuuuuuuuuuuuuuuuuuuuuuuuvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
                wwwwwwwwwwwwwwwwwwwwwwwwwwwwwxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                yyyyyyyyyyyyyyyyyyyyyyyyyyyyyzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
                #############################$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
                %%%%%%%%%%%%%%%%%%%%%%%%%%%%%^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
                &&&&&&&&&&&&&&&&&&&&&&&&&&&&&*****************************
                +++++++++++++++++++++++++++++????????????????????????????
```

Informational error messages encountered during execution of DML statements by CBSV programs are still written as ZE records on the System Control Repository (FILE01).

The error messages are reported and deleted using ZE-PRT or are displayable using DSP01.

The layout of the ZE record is as follows:

Columns	Contents
01–02	ZE
03–22	Organization Control Number (Control 1-2), Employee Number, Labor/History Number
23–24	Sequence Number
25–38	Time and Date
40–43	User Code
46–51	Program/Form Name
52–55	Table Name
56–61	SQLCODE
62–74	Segment Key

Note: *For improved usability, run DSPERR using the Process Confirmation - Ad-Hoc Jobs form (ADHOC).*

New CSL requirements

Although our commit methodology enforces a logical unit of work (LUW) throughout CBSVO/B for update of both online and background (batch) transactions, it is possible for CSL programs to have high volume I-O within a logical unit of work. To ensure these processes remain within your system's configured commit limit, SQLLIM, a new CSL verb—COMMIT-CHECK—has been delivered. Each time a record is updated, a counter (SQLCNT) is incremented. Each time the verb COMMIT-CHECK is executed, it performs a check to see if the SQLCNT > (SQLLIM – 100) and performs a COMMIT when the condition is met. Several core CSL programs have been updated to include a call to the COMMIT-CHECK verb after each update. They are:

ABSTMC
AE-@UK
CYB707
CY710P
CYB710
CYB90B
DEMOY2
F-XREF
HEDUPD
HHMSCR
HHM@UK
KEYDEL
KEYDRH
MAINTI
MASSTR
POPF01
PURGE
QUERY
RELOADW
TCLPAY
TMC@UK
TMCARD
WWWDEL

COMMIT-CHECK is a quick verb and its use does not impact run time performance and is recommended for customer use should there be custom CSL that includes the high volume I-O.

Changes to the installation process

The installation of the relational version has changed in so far as the very first execution of CBSVB, normally the DEMO process, will begin with a relational version of CBSVB. This program together with an EXPORT.10 will be delivered in the install media.

To ensure consistency and limit the RDBMS licenses to 1 per on-line session, we have moved the write of menu records, server checklists and user profiles from program CYBIO to program(s) CBSVO/CBSVOT.

Changes to the administration process

For a relational implementation, the steps to implement a PN fix affecting the data dictionary have changed. This includes adding, changing or deleting tables, fields, data types, data length, or data precision.

☞ *Refer to Application of Temporary Fixes in the Technical Administration documentation for detailed steps.*

Support for bilingual French and English

An international group from Canada, the United Kingdom, and the US worked together to provide a bilingual version of eCyborg (including the Windows and Web user interfaces, Interactive Workforce, and Cognos Impromptu catalogs). Although The Solution Series has supported alternate language functionality for many years, this new functionality is different.

A Canadian company need not implement either an English or French language version of the product; the company's user can elect the language they wish to use when they log in.

This product will be available in Canada to customers who order the French version of the eCyborg product.

Large number support for amounts larger than 9,999,999.99

The Solution Series now allows the storage of accumulator values up to a maximum of 999,999,999.99.

This enhancement addresses errors that have occurred calculating tax in the US when large bonuses have been paid. This enhancement also ensures that paychecks and W2s can be printed correctly. In addition, it significantly reduces the chance of errors in certain countries where the currency and inflation rates have previously led to the system not being able to cope with higher paid employees.

Note: *This change does not attempt to solve the issue of static fields, for example Annual Salary in the LZF segment.*

☞ *Refer to the Using Payroll Administration documentation for details about payment reversals (not allowed for any Net Pay amount over 999,999.99) and online pay calculations (must replace report generator RPT5Z with RPT5ZL).*

☞ *Refer to the Technical Administration documentation for details on the LUC\LUD and XLUC\XLUD pay document formats.*

Technical considerations

Important!

Support for large numbers necessitated a major change in data structure for the 5.2 system. The 5.2 system supports only 6-byte data formats. This guide will instruct you on the procedures you must follow in order to convert your data to the accepted format.

- To display additional digits, a number of new RG instructions, together with additional R5/6 Edit Patterns, have been introduced. Existing RG instructions and R5/6 Edit Patterns will produce the same results as at present.

- The P40OUT is limited to 200 bytes. Although all of the supplied report generators operate safely within this limit, it is possible that overflow may occur within your custom generators if no allowance was made for the maximum counter size. This may result in incomplete data appearing in reports. If you believe that there is a risk that this may happen, we recommend that you run a test pay run using the H2 debug option 'D' in column 13, which will report in the Audit Trail any generators where overflow occurs (without a P20OUT being produced).

- CSL reports are limited to 150 bytes. There is no easy way to determine record length in CSL reports; however, computational fields are not often used in CSL reports. Check your report details carefully.

- If you have custom check generators, you must make a change to the type '9' extract record (change instruction 'FLD087L19' to 'FLD087'). This refers to particular records that are written as part of the payslip RG. The type 9 writes information that gets past through to the Combined Register. If this change is not made, P4CALC will abort on some platforms (z/OS) or the check recon number will not recycle back to the p20 history record on other platforms. Also the name line on the Combined Register will be incorrectly formatted, as will the employee name on the Payment Register.

- Computational field sizes have been increased. If you have custom computational fields, their sizes will be updated when you import your changes via (MAINTI). Data type 3 is no longer supported—Data type 4 is now required.

See also:

- Large Number Changes to Fields *(on page 197)*

Support for communication events triggered from the web client, Interactive Workforce, or the Collaborative Platform

The event functionality was introduced with the 4.5 release of the Administrative Client and allows the automatic generation of emails and letters. A trigger can be set-up and executed via a form (Add, Change or Deletion of a transaction) or a Checklist action (Pause, Finish).

In the 5.2 release, the 'form trigger' event functionality within the Administrative Client is also supported for the Web Client, Interactive Workforce, and applications that use the SQL driver, such as the Collaborative Platform.

The following table shows a comparison between Client-based and the new Server-based events:

Feature	Client-based event	Server-based event
Trigger from all clients (including Checklist events for Administrative and Web clients)	No	Yes
Email user options (such as Preview)	Yes	No
Letter dialog with options to Preview/Queue/Print	Yes	No
Generation of queued mail merge data (downloaded via the use of the Web Application for letters)	No	Yes
Manually execute an event from the toolbar	Yes	No
Option to suppress an event assigned to a form if it is in a checklist	Yes	No

☞ *Refer to the server events installation steps in your eCyborg installation guide for detailed steps for installing and configuring the events server.*

☞ *Refer to the client installation steps in your Solution Series installation guide or to the Optimizing System Features documentation for detailed steps for creating and modifying event triggers.*

Technical considerations

New system file

A new file (FILE51) has been added to The Solution Series to store pending events.

An event can be set-up and identified as a server-based event

- The Solution Series stores form information if a form is updated AND is associated with a server-based event.
- A Windows service called the Event Server is responsible for executing server-based events. Any pending events stored via The Solution Series are processed against server-based events as set-up via the Administrative Client. The processing of server-events will occur via an interval (set by the user) and can be changed by using the Control panel Application for the service.

- A web application is used to download generated letters.

Backward compatibility

▪ Any client-based events you may have on your system are backward compatible with the addition of server-based events. Client-based events will work as before and be triggered and processed within the Administrative Client. You can simply change the client-based event to a server-based event by editing the event configuration and selecting a Server Event checkbox.

▪ The setup is no different except for the addition of the Server Event option; and if checked, certain options are disabled. The default option is set to trigger as a client-based event.

▪ All form trigger functionality is supported for email and letter events, except:
  ▪ the 'Trigger when selected from Toolbar' option
  ▪ the 'Suppress form in a Checklist' option

▪ Some options that rely on client interaction such as the 'User options' shown on the Email Detail tab are disabled when setting up a server-based event (a dedicated server machine sends the actual emails for a server-based event, not the original client machine that triggered the email as with client-based events).

System requirements

▪ A Windows machine is required to execute the service, interact with MAPI compliant email systems and Microsoft Word, so this machine must adhere to the same prerequisites as the Administrative Client.

Application Server

Create FILE51 records for forms that have associated server-based event records. The FILE51 records contain all required event information.
Note:
The event information is generated by The Solution Series server (cbsvo). This causes records to be generated for all client applications (see note).

Event Server
Uses a Windows service to read and process FILE51 event records. Emails are sent directly to recipients and letters are queued for printing. The viewing and printing of these letters are actions performed via the browser client.

Windows 2000 Server
Event Server, containing:
- Web Server
- MAPI Compliant Email Server
- MS Word

Ethernet

Internet/Ethernet

Letters can be printed to any printer defined by the browser client, including network printers

Laser Printer

Browser Client
View-generated letters web application

The administrative client is used to configure server-based events in the same way that client events are configured today

Admin Client

NOTE:
The following clients are supported:
- Administrative Client
- Interactive Workforce
- Web Client
- Collaborative Platform

Configuration suggestions

It is most likely that if you choose to use server-based events for your system, you would only have server-based events set-up within your system. This option removes the need to have Microsoft Word and MAPI set-up for Administrative Client users. Of course, those users who set-up and manage events still need Microsoft Word and MAPI set-up. This option also benefits the user in that all letter-generated events will be managed via the web application and not within the Administrative Client. This may otherwise prove confusing to a user who uses both the Web Client and Administrative Client because he or she would need to look in two areas for batches of queued letters.

Combining client- and server-based events is practicable if you observe the following 'rules':

- Ensure that only client-based events exist for a particular form.
- If both a client-based and server-based event exists for a particular form, profiling must be set-up and used as a filter within the server-based event profiling conditions. This will stop the server-based event from being triggered if using the Administrative Client.

Note: *The only benefit the user gains from a client-based event is with the interactive options 'User Options' for email.*

You can also take advantage of the Profile Conditions functionality to suppress Interactive Workforce form triggers for any chosen events. This is because Interactive Workforce uses one account to connect to The Solution Series. The same can be applied for the Collaborative Platform if you do not want certain events to be triggered from the Collaborative Platform.

Automatic employee number assignment

Due to increasing concern over regulatory and privacy concerns, organizations are finding that they must automate the employee numbering scheme. Historically, employee numbering schemes have gone through a number of stages. At one point an employee number contained a number of identifying characteristics about the employee, including gender, birth date, department, and so forth. As organizations upgrade to more sophisticated HRIS systems, the need to have an employee number contain this type of information has diminished. Still needing a unique identifier, many employers turned to using the employee's government issued identifier (social security number in the United States). Now, due to privacy issues and the proliferation of identity theft, the use of these governmental identifiers for a company's identification of individuals has come under fire. It is now considered best practice to use random numbers as employee numbers. Since this number should now be completely random, it is much easier for the system to generate it.

In addition to the concern for privacy, organizations are also increasingly conducting their applicant processes through the web. To automatically bring web applicants into The Solution Series, each applicant must have an applicant/employee number assigned for tracking purposes within the system. This enhancement allows easy collaboration with third party recruiting services. These services can be used to add applicants to the Applicant Administration module, allowing the automation of 'onboarding' selected candidates as employees.

The Company-To-Rules Cross-Reference For HR 2 form (BX-SCR) holds only one control number text box, which is for the automatic numbering functionality. This feature, when activated, assigns employee numbers when a new applicant or employee is added to the system. The control number allows you to specify whether you want to utilize a shared numbering scheme across all organizations or use unique numbering schemes.

The Automatic Employee Numbering form (AUTSCR) is used to specify settings so that employee or applicant numbers are automatically assigned when a new employee or applicant is added to an organization using one of the following forms:

- Employee Information form (EF-SCR)
- Set Up A New Employee form (NH-SCR)
- Applicant Information form (001SCR)

You can have employee and applicant numbers:

- automatically assigned (the automatic numbering feature is active)
- assigned manually (the automatic numbering feature is inactive)--this is the default setting
- automatically assigned or entered manually (the automatic numbering feature is neutral)

Automatically assigned employee numbers are numeric and always have 10 positions with leading zeros.

During implementation of the automatic numbering feature, enter the starting number with which you wish to begin employee and applicant numbers in the Last Number Used field. Thereafter, the field is updated to indicate the last number assigned to the new employee or applicant and the date on which the number was assigned displays below the field.

Note: *Automatic employee numbering is a 'from this point forward' feature. If you already have employees with employee numbers in The Solution Series, you may manually convert their employee numbers to your numbering scheme at your leisure, or simply utilize the auto-numbering feature for new employees.*

☞ *Refer to the Implementation Essentials documentation for more details of configuring an employee numbering scheme.*

☞ *Refer to Using Human Resource Administration and Using Payroll Administration documentation for details on hiring an applicant or new employee and the impact on this process, depending on the scheme you implement with respect to automatic employee numbering.*

Enhanced setup for the Benefits Recalc Report (85-RPT)

The Benefits Recalc Report (85-RPT) is meant to be run to implement plan or employee changes exclusively. Often, users selected both of the choices (plan or employee changes) resulting in lengthy or invalid runs of this report. The Solution Series now presents a reject message if both options are selected. The user is notified immediately that the selections is invalid and can correct his or her selection prior to running thew report.

☞ *Refer to the Using Benefits Administration documentation for details about this report and its parameter form.*

New Workforce Planning functionality

Workforce planning is a strategic capability necessary for the most successful organizations. The organization is able to compare the current competencies of its workforce to the competencies of the work needed. More importantly, it allows an organization to plan for the future, implement succession planning, and identify career paths along which human capital can be moved to increasingly responsible positions. The ability to track readiness and progress is paramount to the success of the organization. An

organization must identify its key positions and determine if it is ready with replacements should one of these key positions become vacant.

Enhancements in the 5.2 release support these activities and provide the building blocks for the desired result: leveraging the data in the system, turning it into information and ultimately performing the type of analysis needed to project, plan, develop, and hire human capital that can have the capability to maximize the competitive position of the organization.

Three new forms have been added to the system:

- **Career Paths (MCPSCR)**—used to establish career paths that list the succession for up to eight positions using the Career Paths form (MCPSCR). On this form, you can link two career paths together by indicating the career path that would be pursued once the final position of the first career path is achieved.

- **High Potential Employee (HP-SCR)**—used to record a special significance an employee has to the company. For example, an employee may be considered on the 'fast track' or as having high potential. Such distinctions can be recorded on the High Potential Employee form (HP-SCR). The High Potential Designation option list (PM31) used on this form can be populated with designations to meet your business needs.

- **Key Position (MKPSCR)**—used to designate a Position as having a special significance in your organization. For example, a Position may be designated as 'hard to fill' or 'hard to retain'.

☞ *Refer to Using Position Administration for details on designating significance to a position and establishing career paths.*

☞ *Refer to the Using Human Resource Administration for details on designating significance to an employee.*

24/7 Access for online updates during payroll runs

Windows and UNIX platforms only

This feature allows the ongoing use of all areas of the system—even while a payroll is being run. Prior to this release, users executing a selective pay run had to log off the system before the final phase of a payroll (update to the online database). By implementing a 24/7 payroll cycle, along with the behind-the scenes movement of the updatable forms into Inquiry mode, any 'down' time is hidden from the users and is very short termed. Implementing this feature is an option—it is an additional activity—but the 5.2 release includes the components required to make this feature work at your site.

The methodology protects certain data from being updated externally to the pay run process. This feature has also been designed to interface with the Enhanced Payroll Processing and Reporting and the Process Monitor and Report Viewer tools. You must have installed these tools to fully utilize the 24/7 Access feature.

If you elect to implement this feature, your payroll process will be altered. This feature assumes you will be running selective payrolls.

☞ *Refer to the Implementing and Using 24/7 Access documentation for details on the implementation steps required.*

Enhancements to Reporting Administration between 5.0 and 5.1

Reporting Administration has been enhanced to provide additional data and to make updating your custom catalogs much easier.

Catalog enhancements

Reporting Administration is currently made up of four catalogs—General, Benefits, Training Admin and Payroll. These catalogs were designed specifically for typical user groups within the HR domain. While most of the data in The Administrative Solution is contained in one or more of the four catalogs, the 5.2 release offers the major addition of applicant information, as well as information related to grievances and discipline.

Following is a general breakdown by catalog of the content additions and changes.

General Catalog

Employee Additional Contract Information	In 5.0 data mart, added to 5.2 catalog
Employee Emergency Contact Information	In 5.0 data mart, added to 5.2 catalog
Employee Non-Monetary Compensation	In 5.0 data mart, added to 5.2 catalog
Employee Discipline and Grievance	In 5.0 data mart, added to 5.2 catalog
Employee Health and Safety	In 5.0 data mart, added to 5.2 catalog
Applicant Tracking	New
Name and Address	4.5 Enhancement

Benefits Catalog

Employee Additional Contact Information	In 5.0 data mart, added to 5.2 catalog
Name and Address	4.5 Enhancement

Training Admin Catalog

Employee Additional Contact Information	In 5.0 data mart, added to 5.2catalog
Name and Address	4.5 Enhancement

Payroll Catalog

Employee Additional Contact Information	In 5.0 data mart, added to 5.2catalog
Name and Address	4.5 Enhancement

The data reflected on the following forms are included in the catalogs. This list will expand as additional forms and fields are added to The Solution Series.

Employee Additional Contact Information	
EM-SCR	Employee Contact Information

Employee Emergency Contact	
16-SCR	Emergency Contact/Physician

Employee Additional Contact Information	
EM-SCR	Employee Contact Information

Employee Discipline and Grievance	
DA-SCR	Disciplinary Actions
GT-SCR	GrievancesCurrently in DataMart

Employee Health and Safety	
15-SCR	Emergency Medical Information
PR-SCR	Injury Information (RB-01-023 added fields to this existing form)
18-SCR	Physical Exam Schedule/Completed
19-SCR	Physical Exam Test Results

Name and Address	
EF-SCR	Name and Address (General Release 4.5 enhanced name & address)

Applicant	
001SCR	Applicant Information
002SCR	Applicant Name and Address
008SCR	Applicant Contact Information
004SCR	Applicant Preferred Work Environment
006SCR	Applicant Work History
007SCR	Interview Recap
009SCR	Applicant Job References
003SCR	Job or Position Applied For
020SCR	Hired Applicant Pre-Transfer Information

Note: Data from the Applicant Identifying Documentation form (005SCR) is accessible for applicants from the employee folder. This is similar to other forms that are accessible for both applicant companies and employee companies (for example, competency data).

Improved Extract and Load performance - Incremental extracts

History and Labor data has always been extracted from The Solution Series incrementally. Previously, however, employee data could only be extracted in its entirety for inclusion in the Workforce Data Mart. With the 5.2 release, a new option is available: a one-time build

and incremental updates thereafter (although we recommend occasionally rebuilding the full data mart) of the employee data.

Note: *Position Administration data is extracted in its entirety--it is not extracted incrementally.*

When you elect to use an incremental extraction methodology, only information that has changed (add, change, or delete) since the last extraction is incorporated into any extractions performed after the data mart's initial build.

☞ *Refer to the Technical Administration documentation for more information and detailed steps for running incremental data extracts for Reporting Administration.*

Technical considerations

To elect an incremental extraction option, you must activate Reporting Administration on the System Options form (SCOPTS). The COBOL checks this field to see if it needs to write records to a system file, FILE08. When activated, FILE08 is where all changes made to the Employee Database (FILE02) and FILE01are recorded. When the incremental extraction is performed, this file is read for data that must be refreshed in the data mart.

The parameters form for the Reporting Administration extract (R-RSXR) form has been revised to provide a new filter, which allows the RSXRPT Extract Report to extract data in FULL or INCREMENTAL mode.

Note: *Only data from employee level segments is extracted incrementally—not Position Administration or Organization data.*

Modification to EXTRACT PROGRAMS

If you have customized or created additional data mart extracts, you must revise them for use in an incremental extraction. All employee level programs have been modified to process only the segment changing. If an L-segment you need has changed, you must locate that changed segment. Below is the sample code used by some of the delivered extract programs.

```
IF R-RSX-EXTRACT-TYPE EQUAL 'I'      (check if Incremental)
    MOVE W0-15-819 TO L-SEGMENT      (move additional key from change)
    MOVE W0-05-860 TO R-RSX-TARGET   (Target is required for the table update)
    FIND L-SEGMENT STARTING WITH L-SEGMENT-KEY (find segment change by using key)
    PERFORM P200-TABLE-DATA
    GO TO P999-RETURN.
```

Change L-SEGMENT to the segment being changed – G-segment, J-segment, H-segment, F-segment, or P-segment.

☞ *If you have attended the Customizing Reporting Administration class, contact Support— they can provide you with an updated guide. You must attend this class if you wish to customize the data extracts and receive support for your customizations.*

Catalog Upgrade Macros

Our database and Cognos experts have devised a more automated way to update your catalogs with the enhancements we are delivering with 5.2 and upcoming releases—without overwriting the modifications you may have made. That means we deliver the catalogs, then you make your customizations (including: add folders, change the names of folders,

add security user classes, define user class passwords, add alias tables, change the joins, and so forth). When we deliver the next set of catalogs with enhancements and updates, you should not lose modifications.

To achieve this, we delivers two new Cognos macros:

- **Compare macro**
  This macro compares the two catalogs (old and new 'vanilla') and creates the output files for new additions/modifications for the new catalog. Additions/modifications are captured for all catalog tables, columns, joins, all folder levels, fields, prompts, derived fields, and filters.

- **Update macro**
  This macro reads all the output text files (generated by Compare Macro) reflecting the differences between two catalogs and applies the changes to your customized catalog.

☞ *Refer to the Upgrading Reporting Administration to 5.2 documentation for details on performing an upgrade to your Impromptu catalogs.*

Enhancements to Interactive Workforce between 5.0 and 5.1

The 5.2 release of Interactive Workforce offers additional support of IW-enabled workflows. including:

- Employee name and address changes
- Internal job applications
- Employee competencies
- US benefits rollover
- Mid-year enrollments for life event changes

Additionally, the 5.2 release offers a simplified multi-environments installation and implementable single sign-on capabilities.

Password case sensitivity change

As a security enhancement for the 5.2 release, we changed the logic for Interactive Workforce so that the password validation logic is now case sensitive. Previously, it was case insensitive, but the system converted everything to upper case in the database. So, when you upgrade to 5.2, all existing passwords will be treated as upper case and users must type them in that way to gain initial access to the upgraded system. Once a user has accessed the system for the first time, he or she can change the password to lower case or mixed case. The password will be saved in the database in the case entered and the user must type in his or her password with the correct use of case going forward.

Employee name and address changes

The name and address feature in Interactive Workforce allows the HR department to view and acknowledge changes employees make to their name and address information. When an employee changes any information on the Name and Address pages in Interactive Workforce, you can:

- Allow the system to automatically update the employee's permanent record on the Employee Name and Address form (EF-SCR) in the Administrative Solution

OR

- Route all changes to name and address information to the HR department for confirmation

To route changes to the HR department you complete the Workflow Control Table for Administrator form (T933CR). If you do not complete this form, HR will not receive notification of a change to name and address information.

☞ *Refer to the Interactive Workforce: The Administrator's Guide documentation for more details.*

Internal job applications

Interactive Workforce now provides capabilities for displaying position openings. Employees can view and apply for open positions. Managers can view the requisitions for their area, and also see who has applied for openings. An applicant profile is provided showing all competencies for the applicant.

To use the online Requisitions and Applications feature, your organization must have selected 'In use' from the drop down list in the Pos Admin field and selected 'In use with Rejects' in the Req Admin field on the System Options form (TG-SCR).

Each organization (Control 1-2) must have control numbers set up on the Organization-To-Rules Cross-Reference for HR form (AX-SCR). The Position Administration code on this form distinguishes the routing control tables used for the workflow in Interactive Workforce.

Position Administration must also contain all the positions in your organization and the reporting structure for the positions. You can create requisitions only for positions that have been created in Position Administration.

☞   *Refer to the Interactive Workforce: The Administrator's Guide documentation for more details.*

Employee competencies

Employees can now enter and track their own skills and competencies gained from learning experiences. Organizations can choose to have employee entries verified by their managers and/or Administrators. An employee's competencies information also displays to managers if the employee applied for an open position using Interactive Workforce.

When your organization implements Interactive Manager, you give employees the ability to update their professional background and competencies records online. Employees can update their records in the following categories:
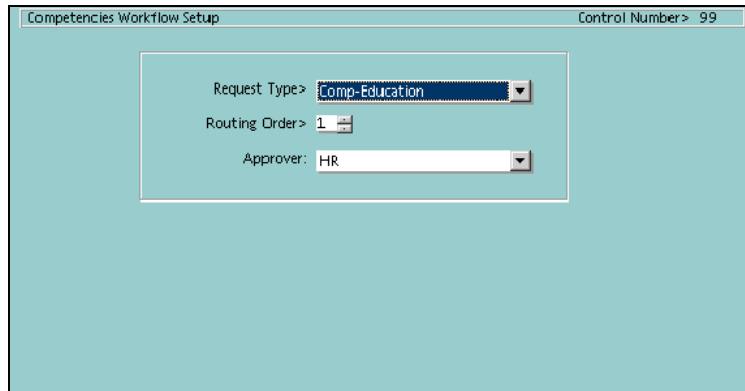
- Certificates, Licenses, and Permits
- Professional Association Memberships
- Formal Education
- Skills and Competencies
- Experience

When an employee creates a new record or modifies an existing record in any of the above categories, they submit the update for approval. Depending the workflow setup for each of the categories, Interactive Workforce routes the record to the employee's manager and/or HR Department. If you set up no workflow, the system automatically updates the corresponding form in the Administrative Solution when an employee uses Interactive Workforce to update his or her competencies.

Note: *The Skills and Competencies form can also be updated when an employee successfully completes a course from the Skills/Competencies panel of the Course Directory form used by Training Administration.*

Using the Competencies Workflow Setup form (T932CR) you determine where you want each category of professional information routed.

Note: *If a professional competencies record has no routing set up, the system automatically approves any requests for that type of record update.*

You can set a different approval routing for each of the categories/types of competencies that appear in the drop down option list. If you do not set up a routing for a request type, the system automatically updates the record in the Administrative Solution. The number in the sequence entry tells Interactive Workforce the order in which you want the request routed.

☞ *Refer to the Interactive Workforce: The Administrator's Guide documentation for more details.*

US benefits rollover

If an employee's benefits choices are identical to the previous year's choices, and the employee does not wish to change anything, a single 'rollover' can be selected during open enrollment.

Open enrollment is a window of time during which all existing employees in the organization have the opportunity to select their benefit coverage options for the following 12 months. Open enrollment takes place during the current coverage period. During open enrollment the employee chooses his or her coverage for the next coverage period, usually a calendar year.

During open enrollment, an enrolled employee can access the following information pages. The system displays a page with three tabs:

- **Welcome to Benefits Enrollment**: Explains the benefits enrollment process in Interactive Benefits.
- **What's new this year:** Contains information for the benefit plans available or the enrollment process that differs from the previous year's information.
- **About the Enrollment Process:** Explains the open enrollment process and contains instructions for benefits enrollment in Interactive Benefits.

When an enrolled employee selects the Benefits Eligibility link on the Navigator, the system displays additional pages with the following benefits information that is specific to the employee:

- **Benefit Choices**: Shows the benefit plans for which the employee is eligible along with a description of each plan. This page may also contain links to additional benefit information and flexible benefits credits if applicable to your organization.
- **Information About You**: Contains employment information about the employee that may affect his or her benefits enrollment.

The remainder of the enrollment process is similar for both newly hired employees and employees who have previously completed the enrollment process.

☞ *Refer to the Interactive Workforce: A Guide to Your Benefits information documentation for more details.*

Mid-year enrollments for life event changes

If an employee experiences a life event change such as marriage, the birth or adoption of a child, or a divorce or legal separation, the employee can modify his or her benefits enrollments to reflect the change. In addition to benefits choices, an employee may need to change personal information such as name, address, and/or beneficiaries. To help update the appropriate information online, Interactive Workforce provides a checklist of information that may have changed as a result of the life event. To access the checklist, the employee selects Life Event from the Navigator pane, selects the appropriate event, and enters the date the event occurred. When the employee clicks Next, the system presents a list of links to pages where the employee can update information.

☞ *Refer to the Interactive Workforce: The Administrator's Guide documentation for more details.*

Single sign-on

Once users successfully sign on to their organization's portal, they will not have to sign on again to Interactive Workforce or to another web application, such as Enwisen. Single sign-on does not apply to the Administrative Solutions.

We highly recommend the assistance of a consultant for this implementation.

As with other enhancements, record keeping or logging of all customization to your original system is very important. By keeping these records using a consistent method, you create an audit trail that can then be reconstructed and applied when you receive upgrades to the software.

Before implementing single sign-on you will need to have installed Interactive Workforce and have the following installed on the server that you are using for single sign on implementation:

- An HTTP server
- A servlet container

Microsoft Internet Information Server and New Atlanta ServletExec are installed as part of the Interactive Workforce installation. If you are using the same server on which you have installed Interactive Workforce to implement single sign on, you will not need to reinstall the HTTP server or servlet container.

☞ *Please refer to the Customer Center to review other HTTP server and servlet containers supported with single sign-on.*

☞ *Refer to the Interactive Workforce: Technical Implementation documentation for more details.*

Simplified multi-environments installation

The autoinstallation executable can now be used to create additional Interactive Workforce environments.

Interactive Workforce is an extension of The Solution Series, and it also supports multiple environments. We recommend a two-server configuration for Interactive Workforce. In the two-server configuration, multiple Interactive Workforce environments run on the web server and communicate with one or more of The Solution Series environments on the Solution Series Application Server.

At the completion of the installation, the following environments may be established:

- Default
- Test
- Production

The Interactive Workforce auto-install program creates the Default environment. The Default environment provides verification of a full Interactive Workforce system. It is strongly recommended that the Default environment be maintained to provide a base on which to install and verify later software releases.

The Default environment also provides a source from which the Test and Production environments are created. The Test and Production environments are created manually by following the directions in the Interactive Workforce installation guide.

Interactive Workforce utilizes a software package, ServletExec from Unify, that requires separate licensing. Each Interactive Workforce environment requires a separate instance of

ServletExec to run. For each Interactive Workforce environment that is intended to be used widely with reasonable performance, a separate ServletExec license is required. Two ServletExec licenses are provided, one for the Test environment and one for the Production environment. The Default environment is usable, but with a limit of 5 concurrent users. Contact your account representative if you need additional ServletExec licenses.

☞   *Refer to your Interactive Workforce installation guide for detailed steps to create multiple environments.*

Web Client enhancements between 5.0 and 5.1

Enhancements to the Web Client include:

- Administrator pages for setting up environments
- Communication event management—letter preview
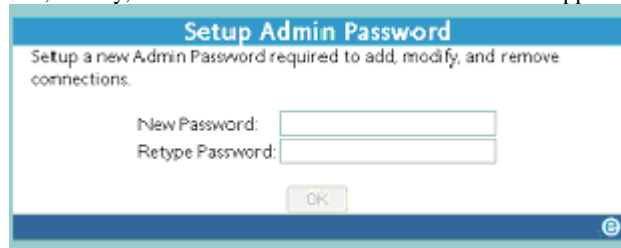
Additionally, the 5.2 release offers client-side XSLT translation for Internet Explorer (for improved performance), enhanced support for the Netscape browser, and support for alternate language/multilingual support (Canadian bilingual version).

Administrator pages on the web

New administration pages allow you to:
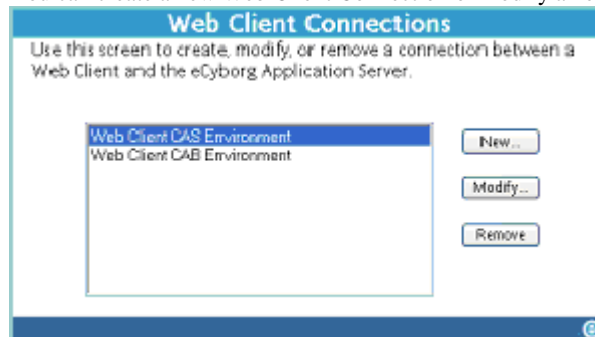
- **Change the administrator password**
  The first time you log on, you must change the password. This password is required to add, modify, and remove connections to the Web Client application.

- **Create a Web Client connection**
  You can create a new Web Client Connection or modify an existing one.

☞ *Refer to your eCyborg or Solution Series installation guide for detailed steps for using the web administration pages.*

Communication events on the web

Communication events are letters or emails that can be triggered automatically within The Solution Series. Communication events include data from the current form or record. For example, a manager may set up a communication event that automatically generates a Salary Increase confirmation letter to an employee when his or her salary is changed on the Salary Assignment/Changes form (40-SCR).

When communication events are set up as Server events, emails and/or letters are triggered by the forms and checklists you use on The Solution Series web client. When you set up a server event, the generated emails are sent automatically, and the generated letters are queued for release.

The Solution Series automatically generates communication event letters and queues them on the server. You access the queued letters through the View Letter Events bookmark on your browser.
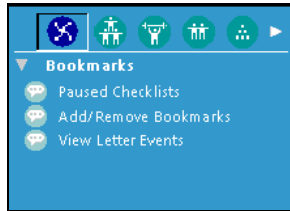
The Queued Letter Events page lists all of the letters queued for printing. The letters are grouped by your ID. To print the letters you must download both the Template (.dat file) and Data (.doc file) files for the letter from the page that displays when you View Letter Events. You use the Microsoft mail merge feature to generate the letter. Downloading and viewing the letters gives you the capability of editing and printing them using the functions and tools in Microsoft Word.

Note: *The mail merge function works differently based on your version of Microsoft Word.*

☞ *Refer to the eCyborg: Using the Web Client documentation for details on using the communication event functionality with the web client.*

Analytics enhancements between 5.0 and 5.1

HR metrics do not live in isolation from the rest of the enterprise. eCyborg Analytics facilitates informed decisions and strategic organizational planning, through the immediate, accessible access to reports and figures it provides. Rather than concentrating on gathering and producing coherent data, HR uses information from analytics to accomplish strategic business goals in all areas of the company.

When HR determines to implement a new training initiative, change benefits, revamp the compensation structure, or outsource a function, it is necessary to identify the expected results and measure the actual results as they relate to the business goal or objective.

Although PowerPlay and some PowerPlay models are delivered with Reporting Administration, eCyborg Analytics consists of Cognos PowerPlay for the web, along with other server-type functionality, such as Upfront. In addition, a major component of Analytics is the addition of more PowerPlay models designed to answer your analytics needs.

With the 5.2 release, a number of Cognos PowerPlay cubes and reports are delivered. Their purpose is to allow the business to track and analyze the impact of its Human Capital. By taking advantage of this analysis, your business can easily pinpoint areas that may require attention, track performance improvement related to various HR initiatives and evaluate the effectiveness of HR programs.

To make business decisions and solve business problems, you must have easy access not only to data and information but also to business intelligence. eCyborg Analytics provides this capability. Analytics allows you to track performance using high-level indicators and drill down to specific areas to determine where performance is not meeting expectation. This enables corrective action to be taken before a crisis situation presents itself. Now you can better manage your business and make the best decision on Human Capital programs that will have the most positive impact on the organization and its employees.

eCyborg 5.2 will deliver four new Power Play models and cubes:

- Turnover Analysis
- Time Away (Absenteeism) Analysis
- Work Related Injury Analysis
- Compensation Change Analysis

The cubes are accessible via the following Cognos applications:

- Power Play (version 7.0 MR2) for client users
- Up Front (version 7.0 MR2) for web users

☞ *Refer to the Using Reporting Administration and Analytics training documentation for details on working with Cognos and the Solution Series catalogs.*

Turnover Analysis

This cube allows the organization to evaluate its turnover from a number of different employee, organizational structure and occupation perspectives. Analysis can be performed on the employee demographics of gender, age, and length of service. Analysis can also be performed against structural elements of the organization such as organization units, jobs and positions. The reasons that employees are leaving can be evaluated against these elements to pinpoint problem areas and area of success.

Examples of Business Questions Answered

- Are separations increasing or decreasing?
- Are voluntary separations increasing or decreasing?
- Are involuntary separations increasing or decreasing?
- What are the reasons for the separations?
- What is the separation rate and trends for various segments of the employee population—gender, age, length of service?
- What is the separation rate and trends for various segments of the organization—org units?
- What is the separation rate and trends for occupations/jobs/positions within the organization?

Time Away (Absenteeism) Analysis

This model and cube allow the organization to evaluate its absenteeism as an employee relations or relationship management metric. The analysis includes all types of time away for which the organization maintains records. Absenteeism can have a profound effect on the performance of the organization. It is often a morale indicator and can be a precursor that foretells future undesirable turnover if corrective action is not taken. An organization cannot functional without its employees and when they are away regardless of the reason, productivity is diminished. It is imperative that the organization maintain vigilance of this area as it can provide valuable clues about employee morale.

Examples of Business Questions Answered

- Is the absenteeism improving or getting worse?
- Is there a particular organizational unit or segment of the organization that is having an absentee problem?
- Does it appear employees in a particular area are taking advantage of paid sick time?
- Does a particular job/position have more absenteeism?
- Is absenteeism increasing or decreasing in the entire organization or for specific segments of the organization?
- Is absenteeism increasing or decreasing for certain occupation groups, job, positions or FLSA classes?
- Does a particular employee demographic (age, gender, length of service) have more absenteeism?

Work Related Injury Analysis

This cube allows the organization to evaluate the severity and frequency of its work related injuries. It is a Work Environment metric. This cube provides the organization with the

ability to evaluate the work related injury history from the perspective of severity and frequency. The alternate drill down paths from Injury Category through to the specific injury description allow the user to evaluate severity along with the additional measures of number of days lost and number of days restricted. This evaluation can be performed for the organization as a whole or down through the organizational structure to pinpoint problem areas and areas that are showing improvement. Employee demographics of age, sex, and length of service can be evaluated in relation to the other dimensions. Occupations groups, jobs and positions add more dimensions for analysis.

Examples of Business Questions Answered

- Is the frequency of work related injuries increasing or decreasing for the organization or segments of the organization?
- Is the severity of work related injuries increasing or decreasing for the organization or segments of the organization?
- Which segments of the organization are having an unacceptable number and/or severity of lost time injuries?
- Which occupation groups, jobs, and positions experience the greatest number of work related injuries?
- What is the frequency and severity of the various types of work related injuries?

Compensation Change Analysis

This analytics cube focuses on monetary compensation. It is a compensation measurement. It is used to analyze changes in the monetary compensation of employees over time. This cube allows the organization to evaluate changes in monetary compensation specific salary, bonuses, commissions and monetary perquisites. It allows the monetary compensation factors to be analyzed from a number of different employee, organizational structure and occupation perspectives. Analysis can be performed on the employee demographics of gender, age, and length of service. Analysis can also be performed against structural elements of the organization such as organization units, jobs and positions.

Examples of Business Questions Answered

- Which segments of the organization receive the greatest/least number and amount of bonuses and commissions?
- Which segments of the organization receive the greatest/least number and amount of monetary perquisites?
- Which segments of the employee population by employee demographic factors receive the greatest/least number and amount of bonuses and commissions?
- Which segments of the employee population by employee demographic factors receive the greatest/least number and amount of monetary perquisites?
- Which segments of the organization are being given the greatest/least number and amount of salary increases/decreases?
- Which segments of the employee population by employee demographic factors are being given the greatest/least number and amount of salary increases/decreases?

Enhancements to the Collaborative Platform between 5.0 and 5.1

The Collaborative Platform was first introduced in release 5.0 of eCyborg, providing the enabling technology to extend the core eCyborg product with various Human Capital Management point solutions (WEB services providing personal finance, travel, and so forth). With the 5.2 release, this powerful feature is enhanced to enable mapping eCyborg data into defined XML schemas supporting both read and write transactions.

☞ *Refer to The Collaborative Platform user documentation for details.*

Technical considerations

Read-Write functionality

Prior to this release the SQL driver only supported an SQL SELECT statement, which allowed read access to Solution Series forms. This release allows update functionality by providing new SQL commands INSERT and UPDATE. These new commands can be used to perform updates to The Solution Series using standard SQL.

☞ *Refer to The Collaborative Platform user documentation for details.*

Problem Notifications, Program Temporary Fixes, and Defect fixes between 5.0 and 5.1

Fixes resulting from customer feedback have been integrated into the system. Feedback has come via Problem Notifications from WebFirst, Program Temporary Fixes posted on CUBBS, and defects identified by the Quality Assurance group.

PTF and PN fixes have been packaged together in the 5.0.1 and 5.0.2 service packs. These service packs have been rolled into the product, and additional PTF and PN fixes have been rolled in.

☞ *For details on the PNs and PTFs included in the 5.2 release, refer to the Listing of PTFs and PNs included in the 51 release excel spreadsheet, available for download from the 5.x Documentation area on the Customer Center.*

North American Regulatory Enhancements between 5.0 and 5.1

Citizenship Code option list (HR05) revised

Important!

This is a global change to The Solution Series, although it was released as a regulatory bulletin for US customers. If you previously performed the operations described in US Regulatory Bulletin RB02-030, you need not convert your data.

The Citizenship Code option list (HR05) has been revised to ensure that it meets the requirements of ISO3166. It also makes the necessary updates to meet Magnetic Media Reporting and Electronic Filing (MMREF) requirements for the US Year End process and US Quarterly Reporting.

The Citizenship Code option list (HR05) is used in a number of forms, both employee and applicant, for specifying a country or citizenship. The option list name has been changed from Citizenship Code option list (HR05) to Employee/Applicant Country Code option list (HR05) to alleviate confusion when using this option list.

US regulatory bulletins between 5.0 and 5.1

The following regulatory bulletins are included in the 5.1 release. Please refer to the relevant RB documentation for more information on any of the RBs.

US Electronic Funds Transfer of Child Support (RB02-018)

Because two states (Illinois and Indiana) currently require that child support payments be submitted using Electronic Funds Transfers (EFT), we generalized the system based on the requirements set forth by the US Office of Child Support Enforcement (OCSE) to accommodate Electronic Funds Transfers (EFT) for child support payments.

In response to the state of Indiana's legislation requiring companies with 50 or more employees to submit child support payments using Electronic Funds Transfer (EFT), modifications have been made to the Child Support Register (6H6H), Child Support ACH Tape (6I6I), and Direct Deposit - Child Support (6R6R) reports, along with the Garnishment Administration form (PO-SCR).

US Supplemental Wage enhancements (RB02-024)

Government guidelines on how to calculate withholding for supplemental wages provide several methods for computing withholding. Many states follow the federal guidelines. These guidelines give you a choice of methods for calculating withholding based on whether supplemental wages are combined with regular wages or paid separately. Employers may combine the supplemental wages with the regular wages or pay the regular wages and supplemental wages separately, for example, issuing a separate payment for a bonus. The Solution Series allows employers to calculate taxes for supplemental wages by three methods: a flat rate method, an aggregate supplemental method, and a table method. The method used is related to whether supplemental wages are combined with regular wages or paid separately.

- If payment consists of regular wages and supplemental wages combined, income taxes will be withheld using the table method or the flat rate method applicable for the taxing authority. This method is delivered on the Tax Authority and Tax Maintenance files.

- If payment consists of supplemental wages only, income tax will be withheld on the supplemental wages by one of the following methods:
    - Flat rate method
    - Aggregate supplemental method
    - Table method

Where the tax authority allows use of the aggregate supplemental taxation method or a flat rate, the flat rate is delivered. With this enhancement, we also deliver a method to override the rates on the Tax Authority and Tax Maintenance files for supplemental wages. Using the Tax Misc2 text box on the Tax Specification Information form (T1-SCR) you can enter a code to override the delivered method. Before overriding the delivered method, however, check the accepted method(s) for the specific tax authority. The entries you make in the Tax Misc2 text box are never altered by a tax update (with the exception of 2AZ) because the fields are left blank on the Tax Authority File (.taf) and Tax Maintenance File (.tmf).

US Equal Employment Opportunity (EEO) and Federal Contractor Veterans' Employment (VETS-100) reporting updates (RB02-022 and RB02-028)

In order to maintain compliance with Equal Employment Opportunity (EEO) and Federal Contractor Veterans' Employment (VETS-100) reporting, we made several modifications and enhancements to existing forms reports, and option lists, and the following have been added to the system:

- VETS-100 Headquarters Location Definition form (VTCSCR)
- VETS-100 Hiring Location Definition form (VT-SCR)
- UVT1 Table Records Batch Format (9V1RPT)
- UVT2 Table Records Batch Format (9V2RPT)
- VETS Establishment option list (EO279)

US INS compliance updates (RB02-032)

To ensure that they comply with INS (U.S. Immigration and Naturalization Service) specifications and include accurate options, updates to the ID/Work Authority option list (HR47), the Identity Established option list (HR49), and the Visa Types option list (HR06) have been made.

These option lists (HR47, HR49, and HR06) are used in the Additional Personal and ID Information form (02-SCR) and the Applicant Identifying Documentation form (005SCR).

US HIPAA Electronic Data Transfer enhancements (RB02-033)

Updates were made to they system to meet the requirements set forth by the Electronic Data Interchange sections of the Administrative Simplification provisions of the Health Insurance Portability and Accountability Act of 1996 (HIPAA).

Standards have been set forth by the Secretary of Health and Human Services (as required by the Health Insurance Portability and Accountability Act of 1996) for the transmission of specific administrative and financial health care transactions, some of which involve data stored within Benefits Administration component. In response to these requirements, we responded with several new forms for setting up company information and capturing employee and dependent information necessary for the transmission of the 834 (Benefit Enrollment and Maintenance) and 820 (Group Premium Payment for Insurance Products) transactions. Professional Services can assist you in creating interface files to enable you to transmit these transactions in the required format.

In addition to updates to existing forms and option lists the following new forms and option lists are delivered in the 5.1 release:

New forms for recording organizational level information
- Additional Benefit Plan Information (UH-SCR)
- Covered Entity Information (HE-SCR)
- Covered Entity Contact Information (HECSCR
- HIPAA Covered Entity Plan Components (HEPSCR)

New forms for recording employee information
- Dependent Custody/Responsibility (DCRSCR)
- Dependent-Related Address Information (DRASCR)
- Medicare/Medicaid Information (MM-SCR)
- Coordination of Benefits Information (COBSCR)—records information for employees or dependents who have enrollment in two benefit plans with similar coverage.

New forms associated with the 834 (Benefit Enrollment and Maintenance) and 820 (Group Premium Payment for Insurance Products) Transaction Sets
- HIPAA 820 Remittance Detail (RD-SCR)
- HIPAA 820 Financial Information (HF-SCR)
- HIPAA - 820 Parameter Control Information (820SCR)
- HIPAA - 834 Parameter Control Information (834SCR)

☞ *Contact Professional Services for assistance in creating interface files to enable you to transmit the 834 (Benefit Enrollment and Maintenance) and 820 (Group Premium Payment for Insurance Products) transactions in the required format.*

New option lists
- HIPAA Entity Identification Codes (BA74)
- HIPAA Identification Code Qualifier (BA75)
- Time Zone (BA76)
- HIPAA Transaction Type (BA82)
- Dependent/Custody/Responsibility Entity ID Code (BA77)
- Dependent Custody/Responsible Entity ID Code Qual (BA78)
- Telephone Qualifier (BA79)
- COB Insurance Product Type (BA83)
- COB Payer Sequence (BA84)
- COB ID Code Qualifier (BA85)
- Federal Admin ID Code Qualifier (BA86)
- Medicare Plan Options (BA68)
- Remittance Type (BA88)
- Remittance Reference ID (BA89)
- Remittance Adjustment Reason (BA92)
- Transaction Handling (BA93)
- Payment Method (BA94)
- Payment Format (BA97)
- DFI Qualifier (BA98)
- Account Type (BAA1)
- HIPAA - 820 Trace Type (BA87)
- HIPAA - 834 Action Code (BA80)
- HIPAA - 834 Purpose Code (BA81)

US California Flat Tax rates (RB03-001)

To accommodate the California flat withholding rate for stock option payments and the various ways of accumulating and reporting imputed income, we created a new form, the

Tax Code Specific HED Overrides form (R2-SCR). This form adds additional flexibility to the payroll tax functions of The Solution Series.

California employers may use a flat withholding rate of 9.3% for stock-option payments that are considered wages, paid on or after January 1, 2002. The 9.3% rate is in lieu of using the standard flat rate of 6% or the withholding tables. With this Regulatory Bulletin, The Solution Series enables you to set up an alternate flat tax to accommodate this situation.

For federal purposes, the value of employer-provided group-term life insurance coverage in excess of $50,000, less any employee after-tax contributions, is considered imputed income and must be included in an employee's earnings. The value of the excess coverage is subject to social security and medicare taxes but is not subject to federal income tax withholding or federal unemployment (FUTA) tax. Therefore, taxable wages must be updated for FICA, but Federal tax is not withheld. The Solution Series accommodates the Federal guidelines by allowing you to update wages but not take withholding tax by using a Tax Frequency of "Report: No Tax Taken".

The above option works for most taxing authorities; however, there are some exceptions. The city of Wilmington, Delaware requires that withholding tax be taken from all imputed income earnings, for example for group term life (GTL). Therefore, taxable wages must be updated and taxes withheld for Wilmington, Delaware. The state of Pennsylvania and some Pennsylvania local tax authorities exclude GTL regardless of the value of the insurance. For these tax authorities taxable wages are not to include GTL nor are taxes to be withheld.

US HIPAA Privacy (RB03-004)

Standards have been set forth by the Secretary of Health and Human Services (as required by the Health Insurance Portability and Accountability Act of 1996) defining the appropriate disclosures of individually identifiable health information (Protected Health Information, or PHI)—information that may involve data stored within Benefits Administration component. In response to these requirements, we are delivering a new form to capture information regarding the disclosure of an employee's or dependent's health information and a report to present this information in a printable format.

The following new forms are delivered with the 5.1 release.

- HIPAA Privacy Authorization/Disclosure form (HADSCR)
- HIPAA PHI Authorization/Disclosure Report (PHIRPT)
- Report Parameters for HIPAA PHI Authorization/Disclosure form (RPHIR)

Note: *The HIPAA Privacy Authorization/Disclosure form (HADSCR) and the fields it contains, and the HIPAA PHI Authorization/Disclosure Report (PHIRPT) (along with the Report Parameters for HIPAA PHI Authorization/Disclosure form [RPHIR] and the fields it contains) should be secured for use by the Privacy Officer and his/her designees only.*

US Tax regulatory bulletins between 5.0 and 5.1

All tax bulletins up through the RB03-010 tax regulatory bulletin, released 24 April 2003, have been incorporated into the 5.2 release. All subsequent tax regulatory bulletins must be applied to the 5.2 release.

US Quarterly regulatory bulletins between 5.0 and 5.1

The latest available Quarterly unemployment regulatory bulletin (Quarter 1, 2003) has been included in the 5.2 release. All subsequent Quarterly bulletins must be applied to the 5.2 release.

Canadian tax updates between 5.0 and 5.1

The following Canadian tax bulletins have been incorporated into the 5.1 release:

RB02-036 and RB02-038

December 11, 2002 - (RB02-038) The federal and provincial changes that became effective January 1, 2003.

December 16, 2002 - (RB02-038) Canada Customs and Revenue Agency (CCRA) issued a modified 'Payroll Deductions Formulas for Computer Programs -77th Edition Effective January 1, 2003'. This entailed changes to tax calculations for the province of Saskatchewan and the territory of Nunavut. This affects only clients running payrolls in Saskatchewan and Nunavut.

Please refer to the relevant RB documents for further details on the contents of each tax bulletin.

C H A P T E R 3

Planning Your Upgrade

In This Chapter

Upgrade considerations

Prerequisites

The software and hardware prerequisites for installing our products vary depending on your platform and the modules you purchased. Some third-party software must be purchased and installed before installing our products. To review hardware and software prerequisites for installing our products, follow these steps:

1. **Access the Hewitt Cyborg home page**
   In the Address area at the top of your browser, type www.hewitt.com/cyborg and then press Enter.

2. **Access the Customer Center**
   At the top of the home page click Customer Center Login.

3. **Log in to the Customer Center**
   Click LOG IN, enter your User name and Password, and then either click OK or press Enter.

4. **Select Product Updates**
   On the left pane of the page, click Product Updates.

5. **Select prerequisites for the Product/Version**
   On the right side of the pane, click the product/version you want to view and their prerequisites.

Deliverables

The following is included:

1	Upgrading The Solution Series from 5.0 to 5.2 (this guide)

Platform	Media
Windows	CD-ROM labeled 'The Solution Series 5.2 for Windows'
UNIX	CD-ROM labeled 'The Solution Series 5.2 for UNIX'
z/OS	CD-ROM labeled 'The Solution Series 5.2 for z/OS'

Timing your upgrades

There are multiple upgrades you will perform in addition to The Solution Series, depending on the functionality in use in your organization such as Reporting Administration and/or Interactive Workforce.

Because the content and architecture of Reporting Administration and Interactive Workforce are dependent upon the content and architecture of The Solution Series, it is important to time your 'go live' date accordingly. Because of architecture enhancements made to the 5.2 version of The Solution Series, Reporting Administration, and Interactive Workforce, once you move into production with The Solution Series 5.2, any pre-5.2

implementation of Reporting Administration or Interactive Workforce will either fail or be problematic.

Plan to go live on all components of your configuration at once, limiting the amount of 'down time' to be experienced by your users.

Server configuration options for Interactive Workforce upgrade

Interactive Workforce requires a version of ServletExec that is mutually exclusive to version 5.2—the version of this software required for 5.2 will not support your earlier versions of Interactive Workforce.

This being the case, by installing version 5.2 of Interactive Workforce on your current live web server you will, in effect, make your production version of the software unavailable to your employees.

You have two options:

- Purchase and configure a 'spare' web server.
  This could be a low-level PC to act as a temporary web server as the test environment need not be powerful. You can use the non-licensed version of ServletExec, which allows up to three concurrent users.

- Use your application server as a web server.
  If you were to use this method, you would:
  1. Copy your production version of The Solution Series to the Application server machine.

  2. Perform your upgrade to The Solution Series.

  3. Install Interactive Workforce on your temporary web server machine

  4. When ready to go live with 5.2, perform the upgrade to Interactive Workforce on your production web server and then switch machines.

Warning - Converting Data (Only applies to pre-5.2 production systems)

Support for large numbers in payroll necessitated a major change in data structure for the 5.2 system. The 5.2 system supports only 6-byte data formats.

To make the data conversion as easy as possible, Distributed Administration components (4- or 5-byte) are provided for installation on your pre-5.2 version system for the purpose of easily transferring your data from pre-5.2 system to the new 6- byte 5.2 system.

Once your data is converted, there is no going back. The data in the 5.2 environment becomes inherently incompatible with the data in your current production system. You can convert the data from your production system as many times as needed before going live on 5.2, but once live, the systems are incompatible.

To maintain historic labor data, you may elect to maintain a system at the current level of your production system. Alternatively, you can convert the data in your archived P20s the same way you convert your live data.

> **Important!**
>
> Distributed Administration programs and functionality are exclusively for use with the upgrade as expressly described herein. No other use of these programs and functionality, or portion thereof, are permitted for any reason whatsoever absent the prior written license grant from Hewitt Associates LLC or Cyborg Systems, Inc. Any non-permitted use shall constitute an infringement of our proprietary rights and subject to vigorous pursuit of rights and remedies available in law and in equity.

Warning - Changes to EXPAND areas

> **Important!**
>
> Because the program areas to store Company and Employee data have been significantly increased, you must remove or modify your existing EXPAND Control Records before upgrading to Solution Series 5.2.

The upgrade process

There are several ways to perform an upgrade, but the process we describe in this guide is believed to be 'best practice'. That is, the most logical approach.

A high-level flowchart of the process is shown below. To move into production, you must at a minimum complete up through Convert Data. The other steps may be done afterwards, but they should still be completed in order for future upgrades to go as smoothly as possible.

Install vanilla non-relational test system

2. Upgrade the Payroll Processing System
- Back up new vanilla system in its entirety
- Review JCL
- Install Distributed Administration
- Review/revise CYBMST overrides
- Review generators and overrides
- Apply modifications and recompile
- Copy 04CALC to CICS region (z/Os)
- Compile sort programs
- Create a P20
- Test Payroll Run

3. Analyze Online System Modifications
- Identify customizations on existing 5.x online system
- Remove unnecessary records
- Update Checklist and Menu records
- Remove user-defined FILE01 table records (Relational)

4. Upgrade the Online System
- Apply customizations (MAINTI)
- Reload
- Convert Citizenship Country codes
- Convert Documentation Management Facility SC55 option list data (Windows and UNIX only)
- Build the data base (relational only)
- Review and revise CBSV overrides
- Expand Employee and Company work areas
- Extract and compile the COBOL programs
- Delete the AREA size record and online executable code
- Build relational System Control Repository table

5. Convert Data
- Create a Data Conversion environment
- Install Distributed Administration and configure for data conversion
- Prepare data for conversion
- Import data into 5.2 system
- Populate tables (relational only)
- Recreate RFT records
- Rebuild Alternate Keys
- Rebuild Phonetic and Employee ID keys

6. Upgrade Online Modifications
- Initialize updates to custom WRITER programs
- Analyze Garnishments
- Delete FILECL32
- Install and configure an updated client
- Add Security records
- Implement security enhancements
- Edit menus to integrate custom programs
- Revise BATCHLs

The following diagram shows the components of The Solution Series and their relationships:

Complete Product Installation Overview
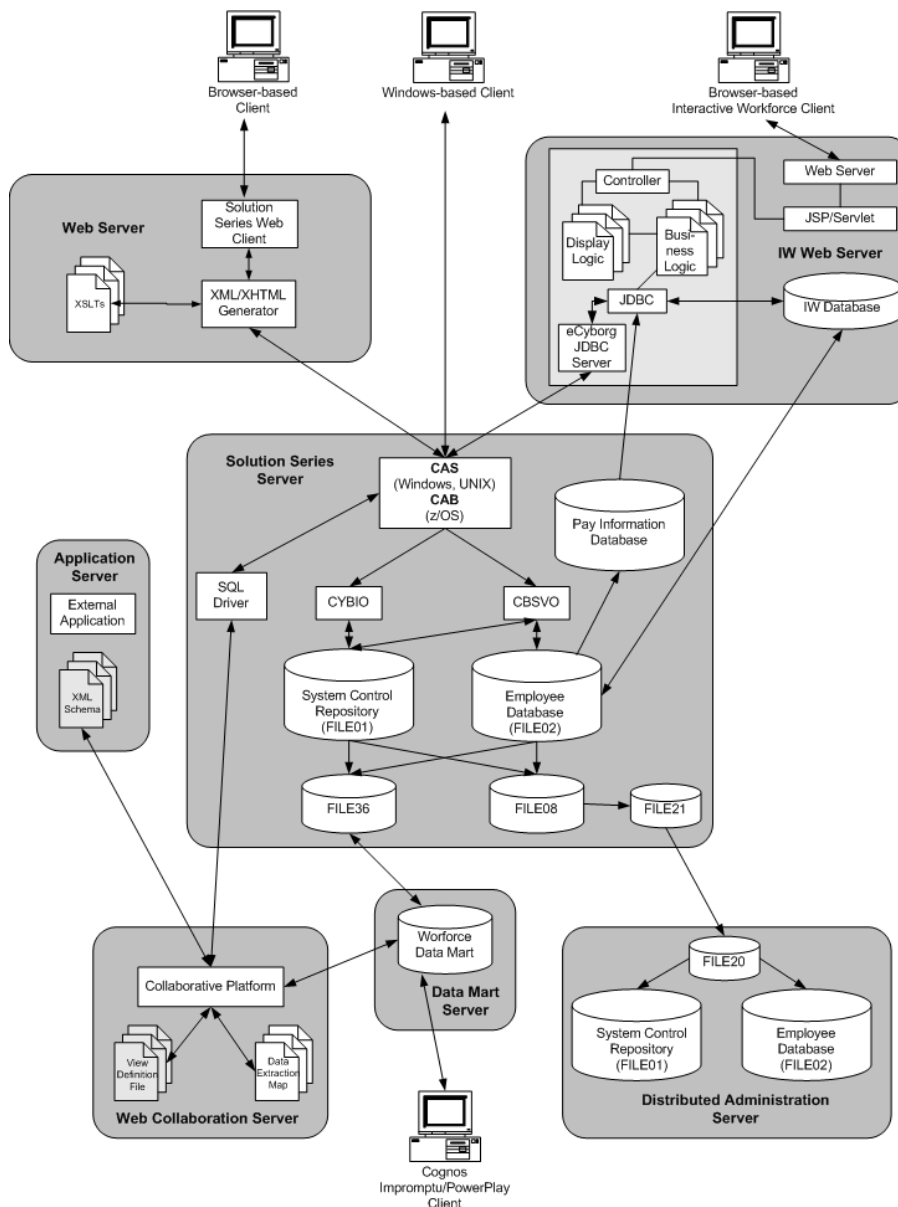
P ART 2

Implementing the 5.2 Release

In This Section

C H A P T E R 4

Implementing the 5.2 Release

In This Chapter

Phase 1: Install The Solution Series 5.2

Important!

Because the program areas to store Company and Employee data have been significantly increased, you must remove or modify your existing EXPAND Control Records before upgrading to Solution Series 5.2.

☞ *Refer to* **Expand Transactions** *(on page 145) for detailed information on the new Expand requirements.*

Perform the following tasks in this phase:

1. Install a vanilla 5.2 non-relational test system.
2. Back up the 5.2 test system.

Task 1: Install a vanilla 5.2 non-relational test system

To install the delivered system, in a non-relational form, follow the steps detailed in the installation documentation for the server platform you require. Once the environment is installed and configured, apply any updates.

Note: *We have rolled in PTFs for the CBSVx programs. If you have any additional overrides you want to bring into the new system, do so during the installation.*

This is the environment onto which you will be implementing your customizations. From this point in the document, we may refer to this system as your 'Test 5.2 system', your '5.2 Test system', or your '5.2 environment'.

☞ *Refer to the eCyborg 5.2 or Solution Series 5.2 installation guide for your platform and complete the steps to perform the installation.*

Task 2: Back up the 5.2 test system

It is very important that you be able to recover from the changes that you make to the 5.2 test system. To protect your test system, back it up **in its entirety** before proceeding with the upgrade.

In addition to all of the program executables, it is important to back up the FILE01, as well as the P20 file that resulted from the test payrun performed when you validated installation.

Phase 2: Install Distributed Administration components on your 5.2 vanilla system

Support for large numbers in payroll necessitated a major change in data structure for the 5.2 system. The 5.2 system supports only 6-byte data formats.

To make the data conversion as easy as possible, we provide a 5.2 (6-byte) version of the Distributed Administration programs. Once installed, you may use the Distributed Administration components (as indicated in this guide) to assist in data conversion and transfer.

We have also (separately) provided (4- or 5-byte) Distributed Administration components for installation on your pre-5.2 version system for the purpose of easily transferring your data from your pre-5.2 version system to the new 6-byte 5.2 system. You will install this version of Distributed Administration later in the process.

Important!

Distributed Administration programs and functionality are exclusively for use with the upgrade as expressly described herein. No other use of these programs and functionality, or portion thereof, are permitted for any reason whatsoever absent the prior written license grant from Cyborg Systems, Inc or Hewitt associates LLC. Any non-permitted use shall constitute an infringement of proprietary rights and subject to vigorous pursuit of rights and remedies available in law and in equity.

Perform the following tasks in this phase:

1. Load the Distributed Administration programs on a PC.
2. Move Distributed Administration files from the PC to the Server.
3. Install Distributed Administration files.

Task 1: Load the Distributed Administration programs on a PC

Read the content of The Solution Series 5.2 Data Conversion CD's HTML Getting Started page very carefully. You must install the 6-byte version of the Distributed Administration programs in your 5.2 environment.

1. **Launch the Distributed Administration 5.2 autoinstallation program (Install_Distrib_Server_v52.exe)**
Follow the onscreen prompts and the Distributed Administration files will be placed on a local PC in a selected folder.

2. **Allocate files (z/OS only)**
The following demonstrates the characteristics of FILE20:

```
Data Set Name  . . . : CYBORG.FILE20

General Data                      Current Allocation
 Volume serial . . . : TSO30A       Allocated cylinders : 4
 Device type . . . . : 3390         Allocated extents . : 1
 Organization  . . . : PS
 Record format . . . : VB
```

```
Record length . . . : 640
Block size  . . . . : 27998        Current Utilization
1st extent cylinders: 4             Used cylinders  . . : 1
Secondary cylinders : 5             Used extents  . . . : 1

Creation date . . . : 2001/05/17
Referenced date . . : 2001/05/17
Expiration date . . : ***None***
```

The following example demonstrates the characteristics of the FILE21:

```
Data Set Name  . . . : CYBORG.FILE21

General Data                       Current Allocation
 Volume serial . . . : TSO30A       Allocated cylinders : 4
 Device type . . . . : 3390         Allocated extents . : 1
 Organization  . . . : PS
 Record format . . . : V
 Record length . . . : 850
 Block size  . . . . : 854         Current Utilization
 1st extent cylinders: 4            Used cylinders  . . : 1
 Secondary cylinders : 5            Used extents  . . . : 1

 Creation date . . . : 2001/04/25
 Referenced date . . : 2001/05/17
 Expiration date . . : ***None***
```

Task 2: Move Distributed Administration files from the PC to the server

(Windows only)

Copy files to correct The Solution Series directories on server

If you look in the directory where the files have been installed, you will find the following directory:

- \Distrib\NT

Under the \NT directory, you will find the following subdirectories:

- \Data
- \Scripts

Copy the contents of those subdirectories into the corresponding subdirectories of The Solution Series environment.

Note: Copy the files from the \Scripts subdirectory to the \Runs subdirectory in The Solution Series 5.2 environment.

(UNIX and z/OS)

1.   Edit FTP job for the correct server name or IP Address
Script used: jftp

Before running this job, you must edit it to use the correct server name or IP Address of the machine where The Solution Series is installed. Open the job in a text editor and add the server name or IP Address to the following line:

```
SET FTPTOSYS=
```

2. Edit FTP job for the correct platform
Script used: jftp

Before running this job, you must edit it to use the proper ftp commands script. Open the job in text editor and add one of the following filenames, depending on which platform you are installing:

z/OS filename: ftpcmds_ds.os2

z/OS relational for DB2: ftpcmds_db2_ds.os2

UNIX filename: ftpcmds_ds.unx

Add the correct ftp command filename to the following line in the jftp job:

```
SET FTPCMDS=
```

Save the changes once complete.

3. Download the files
Script used: jftp

At the command prompt, run the edited jftp job. When entering the command to execute this job, the format should be:

```
jobname username
```

You will be prompted for the password. Review the ftpupload.log, located in the same directory as the install files, for error messages.

You should see 'Job completed'.

Task 3: Install Distributed Administration files

1. Extract, compile, and link Replication Reception program (DSRECV)
Script used: JXDSRECV

This job uses the delivered library files (DSCYBMST for your 5.2 environment) and P9CNVT to extract, compile, and link the Distributed Administration Replication Reception program (DSRECV).

Refer to the delivered JCL for any overrides that may be necessary. Machine parameters are defined in the appendix.

2. Turn on Distributed Administration
Script used: JDSRSET

The JDSRSET job turns on Distributed Administration, making your 5.2 system ready to receive the input from the data conversion environment and apply the converted data.

Phase 3: Analyze and upgrade the Payroll Processing System

Perform the following tasks in this phase:

1. Review CYBMST override files.
2. Revise override files.
3. Review generators and overrides.
4. Apply modifications and recompile the Payroll Processing system programs.
5. Copy 04CALC to CICS region (z/OS).
6. Compile sort programs.

Task 1: Review CYBMST override files

If you have override files, perform this task. As an existing user, you may have several override files - at least one for each of the following:

- COBOL Payroll programs P2EDIT, P4CALC and O4CALC
- COBOL Payroll program P5PRNT
- COBOL Payroll program P9CNVT
- Report and system generators

Review your Payroll COBOL override files

Overrides to the Payroll COBOL programs are applied during the extraction process. The Reader File (P05RDR) contains your overrides to the Payroll COBOL programs (P2EDIT, P4CALC, O4CALC, P9CNVT and P5PRNT).

Task 2: Revise override files

Once you have determined which of your overrides are still valid, edit your override files to remove ambiguous overrides. Save your new files in the same location as your back-up.

The pre-5.2 EXPANDs will not function in extracting programs from the 5.2 CYBMST. You will need to remove or modify your EXPAND control records before upgrading to 5.2. Please refer to "EXPAND area increases" in Appendix A.

Review your Payroll COBOL override files

Overrides to the Payroll COBOL programs are applied during the extraction process. The Reader File (P05RDR) contains your overrides to the Payroll COBOL programs (P2EDIT, P4CALC, O4CALC, P9CNVT and P5PRNT).

Task 3: Review generators and overrides

All Report Generators have been updated in 5.2. Review the Payroll Audit Trail from your production environment to identify additional report generators that you use so that they can be extracted and applied to your P20 master file. Review the delivered report and system generators to determine which generators you want to pull and load to your system.

Review your override file for report and system generators. Overrides to report and system generators are applied during a separate extraction process. The Reader File (P05RDR) contains your overrides.

☞ *Refer to* **5.2 Report Generators** *(on page 170) for details of the Report Generators delivered in 5.2.*

Task 4: Apply modifications and recompile the Payroll Processing system programs

Script used: jxcybmst

This process extracts and compiles the Payroll Processing COBOL program source code from the delivered CYBMST file. The Reader file (P05RDR) varies, depending on the platform. Copy your updated and resequenced override file into the Reader file before performing this operation.

Platform	Location and script
Windows	[Solution Series directory]\Runs**jxcybmst**
UNIX	$[CyborgHome]/runs/**jxcybmst**
z/OS	?HLQ?.JCL.CNTL(**JXCYBMST**)

Task 5: Copy 04CALC to CICS region (z/OS)

After running JXCYBMST, 04CALC needs to be new copied to the CICS region:

```
CEMT SET PROG (04CALC) NEW
```

Note: *You will not be able to see 04CALC in CEMT until it is executed once and loaded into CICS. Once this is done, you will be able to perform the new copy*

Task 6: Compile sort programs

Script used: jcmpsort

Platform	Location and script
Windows	[Solution Series directory]\Runs**jcmpsort**
UNIX	$[CyborgHome]/runs/**jcmpsort**
z/OS	No sort programs delivered as this is handled in the JCL

Review the log to determine if there were any errors.

Note: *Your company may have different sort utilities, but by using the delivered sort programs, we can provide you with better support should you encounter difficulties Client-supplied sort utilities are not supported.*

Phase 4: Analyze and update online modifications

This phase provides one script to upgrade the Payroll Processing components with new CYBMST programs and report and system generators.

Note: *After running any processes, review the output to determine if there were any errors.*

1. Create P20 with your data and new/updated 5.2 RGs.
2. Perform a test Payroll run.
3. Identify your customizations to the online system.
4. Remove unnecessary records and separate the Maintenance Out output into logical files.
5. Update checklist and menu records.
6. Remove user-defined FILE01 table records (Relational).
7. Apply your online customizations.
8. Convert Documentation Management Facility SC55 option list data (Windows and UNIX only).
9. Convert Citizen Country Codes.

Task 1: Create P20 with your existing and new/updated 5.2 RGs

Script used: jcrtp20 (US)
Script used: jcrtp20c (Canada)

Before executing, review each step for input required.

Run this script to:

1. **Extract System Report Generators (RGs)**

2. **Extract other Report Generators (RGs)**

3. **Create 'Empty' Initial P20**

4. **Maintenance to apply RGs from step 1 & 2 to P20 from Step 3**

5. **Extract Additional RGs (Add your custom RGs and overrides here)**

6. **Maintenance to apply RGs (P05T80) and your custom RGs (P05T81)**

7. **PAYXTR ALL (from current 5.2 = copy of production)**

Note: *After running any processes, review the output to determine if there were any errors.*

Platform	Location and script
Windows	[Solution Series directory]\Runs**jcrtp20**
UNIX	$[CyborgHome]/runs/**jcrtp20**

z/OS	?HLQ?.JCL.CNTL(**JCRTP20**)

Important! Do not extract custom report generators and/or method codes in expanded format—only perform the load operation.

Task 2: Perform a test Payroll run

1.   Set up 999999 organization to be paid online

Note: In Canada set up 997777 organization to be paid online.

On The Solution Series system, access the Payroll Run Process Control form (AE-SCR) by making the following selections from the Navigator:

You access this form by selecting:
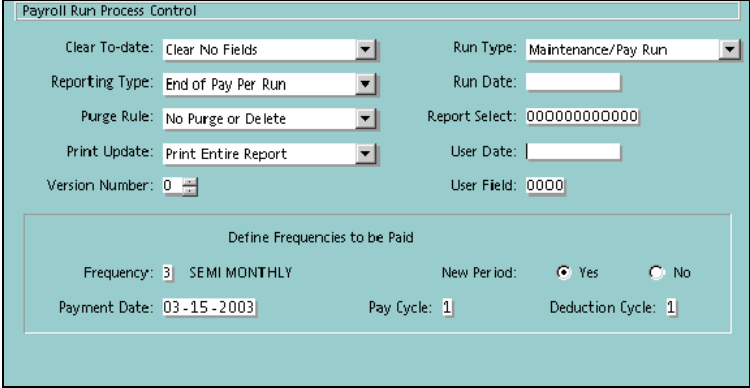
Component:	🏛	Payroll Setup Processing
Process:		Payroll Processing Setup
Task:	🔵	Schedule Payroll Runs

Set up the payroll run parameters, as represented in the graphic, for the following Semi-Monthly pay frequency.

Be sure to set all of the fields as shown on the form. The Run Date field is optional and may be left blank. The Payment Date field must be the current or another relevant date.

In the Frequency field, enter one of the following values then press enter:

▪ '3' for a Semi-Monthly pay run

Log out of The Solution Series.

☞   *Refer to the Introduction to Payroll Administration guide for detailed instructions.*

2.   Update P20IN Batch Master File
Script used: jpayxtr

Test company and employee data from the online Employee Database will be extracted from the online Employee Database and placed in FILE12. This will be the new P20IN master file.

FILE10 (PAYXTR10), which contains time entries and adjustments, is also created. PAYXTR10 becomes input to the JPAYRUN as P05T81.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpayxtr**
UNIX	$[CyborgHome]/runs/**jpayxtr**
z/OS	?HLQ?.JCL.CNTL(**JPAYXTR**)

3. Apply optional transactions
Script used: jpayrun

Note: *If you use Interactive Workforce and have modified the jesspyrn jobstream, run it instead of jpayrun.*

Review the P05RDR file to make sure a 'Y' is in position 18 to ensure the input of the recycle file. The P2EDIT, P4CALC, and P5PRNT programs will be processed.

Include either an S (P05T81 only) or a B (P05T80 and P05T81) in column 19 in the P05RDR file.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpayrun**
UNIX	$[CyborgHome]/runs/**jpayrun**
z/OS	HLQ?.JCL.CNTL.(**JPAYRUN**)

Check your output listings for any anomalies.

4. Perform a maintenance run to create pay history
Script used: jmntrun

To create pay history and labor records, and apply check numbers to the newly created history records on the P20IN Batch Master File, execute the JMNTRUN jobstream. The P2EDIT, P4CALC, and P5PRNT programs will be processed.

Note: *If you modified position 19 in the PO5RDR card, you should remove it from position 19 now.*

Platform	Location and Script
Windows	[Solution Series dirctory]\Runs**jmntrun**
UNIX	$[CyborgHome]/runs/**jmntrun**
z/OS	?HLQ?.JCL.CNTL.(**JMNTRUN**)

5. Update the online Employee Database
Script used: jpaymrg

This process synchronizes the online Employee Database with the updated, final P20 file.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpaymrg**
UNIX	$[CyborgHome]/runs/**jpaymrg**
z/OS	?HLQ?.JCL.CNTL.(**JPAYMRG**)

Task 3: Identify your customizations to the online system

To identify the differences between your custom system and the system originally delivered, perform a Change Control Facility Maintenance Out (MAINTO) operation on your current pre-5.2 version system.

Your upgrade analysis begins by your Solution Series system administrator reviewing the output of a MAINTO operation on your current pre-5.2 version environment. When you run the Change Control Facility Maintenance Out (MAINTO), an output file is produced in FILE10, which you can then use to locate areas of customized code.

Position 80 of each line in this report will contain a change code that indicates the type of difference between the original System Control Repository and the customized System Control Repository.

Change Codes:

Blank	=	Record was added
A	=	Record was added
C	=	Content of the record was changed
D	=	Record was deleted

Program temporary fixes (PTFs) will also display in your MAINTO output.

Notes: *Record types RT, P/S, and F do not have associated PTF numbers..*

Output example:

```
         1         2         3         4         5         6         7         8
1...5....0....5....0....5....0....5....0....5....0....5....0....5....0....5....0
C BA03 03              Employee & One Child                                   D
C BA03 04              Family/EE & Some Dep                                   D
C BA03 06              Employee & Sponsored                                   D
MMNP                 S  Menus15:46:28 07-22N00002815:46:28 07-22              C
MMNP10106            S  Eligibility and &Enrollment                      400  C
MMNP1030509          TRAISC F-CPlan/Fund &Interest Rates                 400  C
MMNP20201            TMBSCR   06Uniform Premium Table                         A
MMNP3000000          II-SCR F  &Injury Information                           C
MMNP3000001          TABSCR F  &Assign Badge                                 C
RQM0102199J30R           QXVSM  M              N                             A
RQM0407200G08R         Y ?X58SPT Y                       !   2Y             A
RQM0412199J07R         N GX261PTC  Y      12      E !D  C0  L    1 P 0       A
T*T991111110002006        RP                                                A
T*Z99111111205C31011001    200K30           P Y9999992001      R            A
T*Z99111111205C31011002    200K30           P Y9999992002      R            A
.
```

Perform a Change Control Facility (MAINTO) operation
Script used: jmainto

To locate any and all revisions to your System Control Repository (Control File; FILE01), you must use the Change Control Facility to perform a MAINTO operation in batch.

Run this job against your existing pre-5.2 version system.

This operation compares your current random System Control Repository (FILE01) with the sequential System Control Repository (DEMO0105) that was originally delivered with your version of the system.

For example, which version of The Solution Series are you running in production right now?

Upgrading from	To	Where to find the DEMO0105 file you need to use
5.0	5.2	delivered with 5.0 vanilla installation
5.0.1	5.2	delivered on CUBBS>Service Pack>5.0.2>I have already applied Service Pack 5.0.1 to my system
5.0.2	5.2	delivered on CUBBS>Service Pack>5.0.2>I have already applied Service Pack 5.0.1 to my system

Any information that does not match with that in the delivered DEMO0105 file is identified on a per-line basis.

Platform	Location and Job
Windows	[Solution Series Directory]\Runs**jmainto**
UNIX	$[CyborgHome]/runs/**jmainto**
z/OS	?HLQ?.JCL.CNTL(**JMAINTO**)

Execute this utility in batch as follows:

INPUT	FILE01	Custom System Control Repository (Control File)
	FILE02	Employee Database (Master File)
	FILE04	Control Record File
	FILE05	[your production version] sequential control (DEMO0105)
OUTPUT	FILE03	Audit/Message File
	FILE10	System Control Repository Change File
EXECUTE	CBSVB	

The control record on FILE04 has the following syntax:

In these positions	Enter	Description
23–28	MAINTO	program name

Control record example:

```
          1         2         3         4         5
1...5....0....5....0....5....0....5....0....5....0....5
                  MAINTO
```

Task 4: Remove unnecessary records and separate the Maintenance Out output into logical files

JCLEAN01, executed from the delivered The Solution Series 5.2 system, compares source code to remove any records in the MAINTO file that are unnecessary because of the inclusion of PTFs in the 5.2 system and system-generated object types; all unnecessary records are removed from the MAINTO output file.

Note: *JCLEAN01 operates on the assumption that all customizations and modifications made to your pre-5.2 version environment were performed according to our programming and naming standards. Customizations that do not follow standards will need to be reworked to standards before they can be loaded into the 5.2 version. Unpredictable results including loss of functionality will occur otherwise.*

☞ *Refer to the Naming Conventions appendix in the Technical Administration documentation for specific naming standards. Programming standards are documented throughout the Cyborg Scripting Language programming documentation.*

To remove unnecessary records and separate the output from your MAINTO operation into logical files, follow these steps:

Script used: jclean01

Run this job from your new Test 5.2 system, against the MAINTO output.

Platform	Location and Job
Windows	[Solution Series Directory]\Runs**jclean01**
UNIX	$[CyborgHome]/runs/**jclean01**
z/OS	?HLQ?.JCL.CNTL.(**JCLEAN01**)

Expected output files:

- **maintocl** (z/OS), **MAINTO.CLN** (Windows, UNIX)
  This is the FILE10 created during the RTPRNT step. It is the 'clean' MAINTO file and should be used in all further MAINTO analysis.

- **recovery** (all platforms)
  This is the FILE03 created during the RTPRNT step. It contains the records removed from the original MAINTO file. This file should be saved in the event any discrepancies are later found.

Both files are subsequent sub-listings of records found in the original MAINTO file.

Note: *Any PTF that has been applied without the new sequence number assigned to the object will be identified and removed. This program will not capture all PTFs—a subsequent manual review of the MAINTO file is necessary.*

☞ *Refer to the Analyzing and Editing the Difference File appendix in the Technical Administration documentation for rules for retaining and removing records form the MAINTO file.*

Task 5: Update checklist and menu records (Windows and UNIX systems only)

Look at the MAINTO from your existing pre-5.2 version environment to check for Enhanced Payroll and Reporting checklist records that have been included in The Solution Series 5.2 system. Duplicate records must be removed from or resequenced in the input file prior to applying your customizations to The Solution Series 5.2 system.

Note: *If you have made extensive modifications to menu records in your pre-5.2 version environment, you will need to re-program those customizations in your new 5.2 environment.*

Warning!

Menu records for HR, Payroll, and some of the Tool items have been redesigned for internationalization. Records now hold an alternate language version of the menu item title. Therefore, you may not be able to use MAINTO to update menu records for these modules.

See also:

■ Changes to Menu Records (*on page 185*)
For the new menu record layout.

1. Check MAINTO of existing system for modifications

Compare the MAINTO FILE10 output of your existing pre-5.2 version system against the following file in The Solution Series 5.2 environment:

- \Data\eprddi05 (Windows)
- /Data/eprddi05 (UNIX)

Look for the following record types:

- MML5 records
- MMNP records

2. Remove duplicate records from the MAINTO file

Records found on the MAINTO file that are also found in eprddi05 should be removed from the MAINTO file. Otherwise, these records will generate errors when you apply the MAINTO file to The Solution Series 5.2 system.

3. Modify the checklist and menu records in MAINTO FILE10

If you have modified the delivered checklists and menu records for Enhanced Payroll and Reporting, you may have to modify the MAINTO FILE10 records to reflect those changes. This may mean creating deletions for the delivered The Solution Series 5.2 records if you resequenced your pre-5.2 version version. Modifications may have to be removed or resequenced.

Task 6: Remove user-defined FILE01 table records (Relational only)

Because there are some relational elements that will be applied to the system before the relational database has been rebuilt, this is the point where you must manually remove any user-defined FILE01 table records (records that begin with 'T', 'U', 'V', 'W', and/or 'X'. If not removed, these records may result in a failed Maintenance In process. Extract your user-defined FILE01 table records to a file, which will be imported later.

Task 7: Apply your online customizations

Having run your MAINTO operation, then reviewed and edited your MAINTO output, you can now apply your 'clean' MAINTO file as input to the MAINTI operation against the 'vanilla' 5.2 system.

1. Perform a Change Control Facility (MAINTI) operation
Script used: jmainti

To apply your customizations to your 5.2 System Control Repository (Control File; FILE01), you must use the Change Control Facility to perform a MAINTI operation in batch. The control record on FILE04 has the following syntax:

In these positions	Enter	Description
23–28	MAINTI	program name

Control record example:

```
         1         2         3         4         5
1...5....0....5....0....5....0....5....0....5....0....5
                 MAINTI
```

Note: *FILE05 is the input file. It must either contain the content of your 'clean' MAINTO file, or you must redirect the job to use your 'clean' MAINTO file as the input file.*

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jmainti**
UNIX	$[CyborgHome]/runs/**jmainti**
z/OS	?HLQ?.JCL.CNTL.(**JMAINTI**)

Check your audit messages. Reconcile any errors before continuing the upgrade process.

If necessary, restore the FILE01 from the 5.2 'vanilla' back up you performed after you installed your 5.2 test system. Fix any problems you identify by editing FILE05, then rerun the MAINTI against the restored FILE01.

Task 8: Convert Document Management Facility SC55 option list data (Windows and UNIX only)

Program used: CVSC55

Prior to this release, the option list used by the Document Management Facility (SC55) used the area reserved for the alternate language to store the document type and default

path for each type of document. In order to use the alternate language area as it was designed (for the alternate language), the document type and path have been moved to a separate line (identified as D00). Since this involves a change to a delivered option list, a conversion program must be run on Windows and UNIX platforms to convert the SC55 option list to the new format.

Perform this step only if your MAINTI indicated that your SC55 (Document Definitions) option list is different than that originally installed on your pre-5.2 vanilla system.

CVSC55 can be run online or in batch. The program requires no parameters.

Note: *This program need only be run once, but accidentally running this program more than once will not affect the outcome.*

Task 9: Convert Citizen Country Codes

Important!

If you previously performed the operations described in US Regulatory Bulletin RB02-030, skip this task. If you have not done so, review the documentation available on the Customer Center for this regulatory bulletin and apply this fix.

The Citizenship Code option list (HR05) has been revised to ensure that it meets the requirements of ISO3166. It also makes the necessary updates to meet Magnetic Media Reporting and Electronic Filing (MMREF) requirements for the US Year End process and US Quarterly Reporting.

The Citizenship Code option list (HR05) is used in a number of forms, both employee and applicant, for specifying a country or citizenship. The option list name has been changed from Citizenship Code option list (HR05) to Employee/Applicant Country Code option list (HR05) to alleviate confusion when using this option list.

Phase 5: Build the Database (Relational Environments only)

Note: *If you do not want to have a relational environment, skip this phase and go on to* **Phase 6** *(see "Phase 6: Upgrade your CBSV Processing System" on page 108).*

Perform the following tasks in this phase:

1. Back up your System Control Repository (FILE01).
2. Export F1 and FTM records.
3. Compile RDBPGM0.
4. Execute the case tool.
5. Compile and link the RDBPGM1 (UNIX only).
6. Create the database/tablespaces, tables, index, and views.
7. Pre-compile, compile, and link RDBPGMA through RDBPGMH.
8. Extract and compile relational 04CALC.
9. Reorganize the System Control Repository (FILE01) (Indexed).
10. Repopulate the Employee Database (FILE02).
11. Recompile CBSVB (Relational only).

Task 1: Back up your System Control Repository (FILE01)

For relational, the output (FILE 10) is the backup of FILE01 and will be needed later in JREBUILD in Task 1 of Phase 7.

To run BACKEM, execute CBSVBT as follows:

INPUT	FILE01	System Control Repository
	FILE02	Employee Database
	FILE04	Control Record File
OUTPUT	FILE03	Audit/Message File
	FILE10	Sequential backup file
EXECUTE	CBSVBT	Program

The control record on FILE04 has the following syntax:

In these positions	Enter	Description
23–28	BACKEM	program name

Task 2: Export F1 and FTM records

To export the F1 and FTM (RFT and RFM) records from the System Control Repository, execute the JEXPORT jobstream.

Script used: jexport

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jexport**
UNIX	$[CyborgHome]/runs/**jexport**
z/OS	?HLQ?.JCL.CNTL.(**JEXPORT**)

Check the audit message file for any errors.

Task 3: Compile RDBPGM0

Script used: jcmprdb0

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jcmprdb0**
UNIX	$[CyborgHome]/runs/**jcmprdb0**
z/OS	?HLQ?.JCL.CNTL.(**JCMPRDB0**)

Task 4: Execute the case tool

1.   Modify the control record in jcrtpgms (Windows and UNIX only)

Modify the control record in jcrtpgms to include the datafile path, database connect string, and the tablespace indicator to uniquely identify this environment.

2.   Execute the case tool

Script used: jcrtpgms

Platform	Location and script
Windows	[Solution Series directory]\Runs**jcrtpgms**
UNIX	$[CyborgHome]/runs/**jcrtpgms**
z/OS	?HLQ?.JCL.CNTL.(**JCRTPGMS**)

Review the log to determine if there were any errors.

Task 5: Compile and link the RDBPGM1 (UNIX only)

Script used: jcmprdb1

To compile the program RDBPGM1, execute the JCMPRDB1 jobstream.

RDBPGM1 creates the tablespaces, tables, indexes, and views needed to support the relational version of The Solution Series.

Task 6: Create the database/tablespaces, tables, index, and views

Note: If you have a relational environment, you must drop your 5.2 (vanilla) database and rebuild the database so it includes your customizations.

Script used: jcrtcyb

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jcrtcyb**
UNIX	$[CyborgHome]/runs/**jcrtcyb**
z/OS	?HLQ?.JCL.CNTL.(**JCRTCYB**)

Task 7: Pre-compile, compile, and link RDBPGMA through RDBPGMH

The following table lists and describes each subroutine generated by the CASE tool:

Program	Subroutine Description
RDBPGMA	Inserts a new row in a table
RDBPGMB	Selects data from a row in a table and passes it to CBSV
RDBPGMC	Updates values in an existing row
RDBPGMD	Deletes an existing row from a table
RDBPGME	Called when a PAYMRG 171 process is run; removes all rows from all tables in preparation for reinsertion of data from the P20 file; also disables then re-enables all indexes (where applicable)
RDBPGMF	Called when a PAYMRG 222 process is run; deletes all rows from the tables that belong to the organizations being paid
RDBPGMG	Cursors through the database and rebuilds the IDX records on the database
RDBPGMH	Provides segment and segment key length for each segment, and location of date and date type within each segment; the link between the database and The Administrative Solution

z/OS

Scripts used: jsqlcomp, jcomprdbh

Platform	Location of Scripts
z/OS	?HLQ?.JCL.CNTL.(**[jobname]**)

1. **Pre-compile, compile, and link the programs RDBPGMA through RDBPGMG**
   Script used: JSQLCOMP

2. **Pre-compile, compile, and link RDBPGMH**
   Script used: JCOMPRDBH

Windows and UNIX

Pre-compile, compile, and link the RDBPGM subroutines.

Script used: jcmpsubr

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jcmpsubr**
UNIX	$[CyborgHome]/runs/**jcmpsubr**

Check the audit message file for any errors.

Task 8: Extract and compile relational O4CALC

To extract COBOL program O4CALC from CYBMST, compile the program, and link the machine-specific subroutines, execute JXO4CALR.

Script used: jxo4calr

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jxo4calr**
UNIX	$[CyborgHome]/runs/**jxo4calr**
z/OS	?HLQ?.JCL.CNTL.(**JXO4CALR**)

Task 9: Reorganize the System Control Repository (FILE01) (indexed) or synchronize FILE01 and TABLE01 (relational)

Important!

This step is required for relational installations and is optional for indexed installations.

In order for the upcoming PULL processes to function properly, FILE01 must be updated and TABLE01 must be in synch with FILE01.

Script used: jebuild

Platform	Location and Script
Windows filepath	[Solution Series directory]\Runs**jrebuild**
UNIX	$[CyborgHome]/runs/**jrebuild**
z/OS	?HLQ?.JCL.CNTL.(**JREBUILD**)

Check the audit message file for any errors.

Task 10: Repopulate the Employee Database (FILE02)

Important!

This step is required for relational installations and is optional for indexed installations.

To run PAYMRG and repopulate FILE02 and the associated Tables after the relational database has been rebuilt. This is also required for the 'PULL' of the CBSV code described in Phase 3. The P20 file used was created earlier in this chapter.

Script used: jpaymrg

Platform	Location and Script
Windows	[Solution Series dirctory]\Runs**jpaymrg**
UNIX	$[CyborgHome]/runs/**jpaymrg**
z/OS	?HLQ?.JCL.CNTL.(**JPAYMRG**)

Task 11: Recompile CBSVB (Relational only)

Script used: jcmpcvbr

During the base installation of The Solution Series 5.2 relational version, a CBSVB was pulled and compiled. You must now recompile (not pull and compile) the CBSVB program so you can use the newly defined database and RDBPGM modules. This recompiled version of CBSVB is required in order to further extract and compile COBOL programs in the next Phase.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jcmpcvbr**
UNIX	$[CyborgHome]/runs/**jcmpcvbr**
z/OS	?HLQ?.JCL.CNTL.(**JCMPCVBR**)

Check the audit message file for any errors.

Phase 6: Upgrade your CBSV Processing System

Perform the following tasks in this phase:

1. Review override files.
2. Revise override files.
3. Expand employee (AREA2) and/or company (AREA4) work areas.
4. Extract and compile the COBOL programs to include the new AREA2-BOTH and AREA4-BOTH values.
5. Delete the AREA size record and online executable code.

Task 1: Review override files

As an existing user, you may have several override files—at least one for each of the following:

- COBOL online program CBSVO
- COBOL online trace program CBSVOT
- COBOL batch program CBSVB
- COBOL batch trace program CBSVBT

Review your online COBOL override files

Overrides to the CBSV COBOL programs are applied during the extraction (CBSVB) process. The override file contains your overrides to the CBSV COBOL programs (CBSVO, CBSVOT, CBSVB, and CBSVBT).

Task 2: Revise override files

Once you have determined which of your overrides are still valid, edit your override files to remove unnecessary overrides.

Note: *COBOL PTFs (identified by a value in column 77–80) must be removed from your override file.*

Resequence your overrides to match the source programs in CBSV and save your new override files where your backups reside.

Task 3: Expand employee (AREA2) and/or company (AREA4) work areas

This task shows you how to move the work area expansions from your existing environment into the new one, and rebuild the CBSV.

1.   Obtain expansion amounts from your pre-5.2 version environment

Log into your pre-5.2 version environment and identify the work area expansions by performing the following steps:

1. Access the Expand Areas in CBSV Programs form. You access this form by selecting:

Component:		Development Tools
Process:		System Operations
Task:		Expand Program Memory

The Expand Areas in CBSV Programs form is displayed:

```
Expand Areas In CBSV Programs

  ****** SOLUTION SERIES    VERS 5.2.0 ****
  CBSVB  WAS PULLED AT  07:06:00 12-15 XXXX
  CBSVBT WAS PULLED AT  07:06:01 12-15 XXXX
  CBSVO  WAS PULLED AT  07:06:01 12-15 XXXX
  CBSVOT WAS PULLED AT  07:06:01 12-15 XXXX

       LAST CHANGED AT  10:50:00 01-22 S.O.

  AREA1    AREA2    AREA3    AREA3    AREA4   DATABASE
  BOTH     BOTH     BATCH    ONLINE   BOTH    COMMIT
  00000    19656    09319    09319    24192   00000
  00000    19656    09319    09319    24192   01000
           _____    _____    _____    _____   _____
           19656    09319    09319    24192   01000

  An AREA 2 expand value of 19656 will allow for an Employee size of 24957.
  An AREA 4 expand value of 24192 will allow for a  Company  size of 32271.
```

2. Take note of the expanded values. These values must be changed in the new 5.2 environment to match.

2.   Apply expansion amounts to your new, updated 5.2 environment(s)

Log into your new 5.2 environment and perform the following steps to expand the work areas:

1. Access the Expand Areas in CBSV Programs form (EXPAND). You access this form by selecting:

Component: Development Tools
Process: System Operations
Task: Expand Program Memory

The Expand Areas in CBSV Programs form is displayed:

```
Expand Areas In CBSV Programs

  ****** SOLUTION SERIES    VERS 5.2.0 ****
  CBSVB  WAS PULLED AT  07:06:00 12-15 XXXX
  CBSVBT WAS PULLED AT  07:06:01 12-15 XXXX
  CBSVO  WAS PULLED AT  07:06:01 12-15 XXXX
  CBSVOT WAS PULLED AT  07:06:01 12-15 XXXX

       LAST CHANGED AT  10:50:00 01-22 S.O.

  AREA1    AREA2    AREA3    AREA3    AREA4   DATABASE
  BOTH     BOTH     BATCH    ONLINE   BOTH    COMMIT
  00000    19656    09319    09319    24192   00000
  00000    19656    09319    09319    24192   01000
           _____    _____    _____    _____   _____
           19656    09319    09319    24192   01000

  An AREA 2 expand value of 19656 will allow for an Employee size of 24957.
  An AREA 4 expand value of 24192 will allow for a  Company  size of 32271.
```

2. Enter the total expanded amount for the employee in AREA2-BOTH. This should be the same amount you expanded to for EMPLOYEE in the the Payroll (P4CALC and O4CALC) overrides.
3. Enter the total expanded amount for the company in AREA4-BOTH. This should be the same amount you expanded to for PAYER in the Payroll (P4CALC and O4CALC) overrides.
4. Press Enter.
5. Log out.

Task 4: Extract and compile the COBOL programs to include the new AREA2-BOTH and AREA4-BOTH values

Extract and compile CBSVO, CBSVOT, CBSVB, and CBSVBT using PULL.

z/OS

Scripts used: jcbsvb, jcbsvbt, jcbsvbo, jcbsvot

Platform	Location of Scripts
z/OS	?HLQ?.JCL.CNTL.(**[jobname]**)

1.   Extract and compile CBSVB
Script used: JCBSVB

2.   Extract and compile CBSVBT
Script used: JCBSVBT

3.   Extract and compile CBSVO
Script used: JCBSVO

4.   Extract and compile CBSVOT
Script used: JCBSVOT

Windows and UNIX

Scripts used: jpulcvn jpulcvr, jcmpcvn, jcmpcvr

Platform	Location and Script
Windows	[Solution Series directory]\Runs**[jobname]**
UNIX	$[CyborgHome]/runs/**[jobname]**

1.   Extract and compile relational Solution Series programs
Script used: JPULCVN (non-relational)
Script used: JPULCVR (relational)

2.   Compile relational Solution Series programs
Script used: JCMPCVN (non-relational)
Script used: JCMPCVR (relational)

Check the audit message file for any errors.

Task 5: Delete the AREA size record and online executable code

To do this, run the ZX-DEL program. You run this program in batch as follows:

INPUT	FILE01	System Control Repository
	FILE02	Employee Database
	FILE04	Control Record File
OUTPUT	FILE03	Audit/Message File
EXECUTE	CBSVB	

The control record on FILE04 has the following syntax:

In these positions	Enter	Description
23–28	ZX-DEL	Name of the program

This program deletes the ZXCYB88W record (or the ZXCYB88M record for all non-PC platforms), as well as any online executable code.

A new record, ZXCYB88W or ZXCYB88M as appropriate, will be created on the Employee Database the next time the COBOL programs are executed.

Phase 7: Build the database equivalent of the non-relational FILE01—Relational Environments only

Perform the following tasks in this phase:

1. RECALC, RETYPE, and RELOAD all Calculation option lists, report layouts, and CSL programs in The Solution Series.
2. Populate RDBMS tables (relational only).
3. Load the user-defined Table records into The Solution Series.

Task 1: RECALC, RETYPE, and RELOAD all Calculation option lists, report layouts, and CSL programs in The Solution Series

Script used: jdemo023

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jdemo023**
UNIX	$[CyborgHome]/runs/**jdemo023**
z/OS	?HLQ?.JCL.CNTL(**JDEMO023**)

Check the audit message file for any errors.

Task 2: Populate RDBMS tables (Relational only)

Script used: jpopf01

To synchronize and populate the System Control Repository and relational tables for option lists and specific application tables (Cx through Xx), execute the JPOPF01 script.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpopf01**
UNIX	$[CyborgHome]/runs/**jpopf01**
z/OS	?HLQ?.JCL.CNTL.(**JPOPF01**)

Check the audit message file for any errors.

Task 3: Load the user-defined Table records into The Solution Series

Script used: jmainti

To apply the user-defined FILE01 table records removed in Phase 4, Task 6, use the Change Control Facility to perform a MAINTI operation in batch. Be sure to change the FILE05 name to match the file created in *Phase 4, Task 6* (see "Task 6: Remove user-defined FILE01 table records (Relational only)" on page 101).

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jmainti**
UNIX	$[CyborgHome]/runs/**jmainti**
z/OS	?HLQ?.JCL.CNTL.(**JMAINTI**)

Check the audit message file for any errors.

C H A P T E R   5

Convert Data

In This Chapter

Warning - Converting Data (Only applies to pre-5.2 production systems)

Support for large numbers in payroll necessitated a major change in data structure for the 5.2 system. The 5.2 system supports only 6-byte data formats.

To make the data conversion as easy as possible, Distributed Administration components (4- or 5-byte) are provided for installation on your pre-5.2 version system for the purpose of easily transferring your data from pre-5.2 system to the new 6- byte 5.2 system.

Once your data is converted, there is no going back. The data in the 5.2 environment becomes inherently incompatible with the data in your current production system. You can convert the data from your production system as many times as needed before going live on 5.2, but once live, the systems are incompatible.

To maintain historic labor data, you may elect to maintain a system at the current level of your production system. Alternatively, you can convert the data in your archived P20s the same way you convert your live data.

Important!

Distributed Administration programs and functionality are exclusively for use with the upgrade as expressly described herein. No other use of these programs and functionality, or portion thereof, are permitted for any reason whatsoever absent the prior written license grant from Hewitt Associates LLC or Cyborg Systems, Inc. Any non-permitted use shall constitute an infringement of our proprietary rights and subject to vigorous pursuit of rights and remedies available in law and in equity.

Overview of data conversion process

In the following phases of this upgrade, you will perform a data conversion and migrate your data over to the new 5.2 system. The following jobs reflect the activities starting in Phase 3, Task 1: Preparing your data (pre-5.2 data conversion environment), which is discussed in detail later is this chapter:

1. N99 PAYRUN
2. PAYXTR
3. PAYRUN
4. MNTRUN
5. GENONLY P20 Creation

The following diagrams illustrate the data conversion for the 5.2 system:

FILE01   FILE02   CONVERT.05

CBSV

JMAINTI to update pre 5.2 system

FILE01   FILE02   FILE05 - CONVERT

CBSV

JCONVERT to pull comment COBOL source for target environment and compile it

FILE01   FILE02

CBSV

JCONVP20 to execute FIXP20 and CRTUDS CSL programs

FIXP20.10

SORT

FIXP20.05

Full P20 from current Production system

CONVERT

Audit report   Validated Production P20 file

To be used as input to Data Conversion process

Phase 1: Create pre-5.2 Data Conversion environment

To protect your Production pre 5.2 system and its data, copy it **in its entirety** before proceeding with the upgrade. Use this copy of your production system to perform your data conversions.

We will now refer to this environment as your 'pre 5.2 data conversion system'.

Perform this operation prior to a full payroll run on your Production system. By doing so, you will be able to perform parallel payroll test runs on your Production and pre 5.2 system, as well as your new 5.2 test system.

Having set up this environment, you may update the data as many times as you wish. Once you have updated the data in this data conversion environment, perform the steps in *Phase 3: Move data into your new 5.2 environment* (on page 127) again. You may then run another parallel payroll test and/or go 'live' into production.

Phase 2: Install Distributed Administration components on your pre-5.2 data conversion system

Support for large numbers in payroll necessitated a major change in data structure for the 5.2 system. The 5.2 system supports only 6-byte data formats.

Important!

If you are currently using Distributed Administration in your pre-5.2 version environment, skip this phase.

To make the data conversion as easy as possible, we provide (4- or 5-byte) Distributed Administration components for installation on your pre-5.2 version system for the purpose of easily transferring your data from your pre-5.2 version system to the new 6-byte 5.2 system.

Perform the following tasks in this phase:

1. Load the Distributed Administration programs on a PC - pre-5.2 system.
2. Move Distributed Administration files from the PC to the Server.
3. Install Distributed Administration files.

Task 1: Load the Distributed Administration programs on a PC - pre-5.2 system

Read the content of The Solution Series 5.2 Data Conversion CD's HTML Getting Started page carefully. You must install the 4- or 5-byte version of the Distributed Administration programs (dataconv_v52.zip) in your pre-5.2 version environment.

1. **Click on step 1 'Install 6-byte Distributed Administration'**
   Launch the Distributed Administration 5.2 autoinstallation program (Install_Distrib_Server_v52.exe) .

   Follow the onscreen prompts and the 5.2 Distributed Administration files will be placed on a local PC in a selected folder.

2. **Click on step 2 'Install updates for specialized 4-or 5-byte data conversion'**
   Launch the zip file (dataconv_v52.zip).

   The zip file contains several files and scripts that will need to overlay some of the version 5.2 Distributed Administration files making this implementation specific to data conversion.

   Load the files from the zip file into the same directory as the base conversion. If you loaded the conversion information from Step 1 using the default directory structure, all the information is now in *C:\[sol series dir]\Distrib*. Unzip the new information in the Distrib directory to automatically create a new directory named DataConv.

3. **Unzip the contents of the zip file (dataconv_v52.zip)**

Opening the zip file in the Distrib directory creates the DataConv folders with its subordinates. Elements within DataConv must be copied to existing folders within the initial structure to accomplish the overlay. The following table itemizes those elements with their current and new locations:

Platform	Copy the following:	From:	To:
Windows	jdspack.bat jdstrib.bat jxdspack.bat jxdstrib.bat jconv02.bat	[Solution Series directory]\Distrib\ DataConv\Scripts\NT	[Solution Series directory]\Distrib\ NT\Scripts
UNIX	jdspack jdstrib jxdspack jxdistrib jconv02	[Solution Series directory]\Distrib\ DataConv\Scripts\UNIX	[Solution Series directory]\Distrib\ UNIX\Scripts
z/OS	jdspack jdstrib jxdspack jxdistrib jconv02	[Solution Series directory]\Distrib\ DataConv\Scripts\zOS	[Solution Series directory]\Distrib\ zOS\Scripts

Review any fixes on CUBBS for scripts and data components that may have changed for the 5.2 conversion. Any fixes will need to replace the existing corresponding scripts or components in the DataConv directory.

Task 2: Move Distributed Administration files from the PC to the server - pre-5.2 system

Windows only

Copy files to correct The Solution Series directories on server

If you look in the directory where the files have been installed, you will find the following subdirectory:

- \DataConv

Under the \DataConv directory, you will find the following subdirectories:

- \Data
- \Scripts

Copy the contents of those subdirectories into the corresponding subdirectories of The Solution Series environment.

Note: *Copy the files from the \Scripts subdirectory to the \Runs subdirectory in The Solution Series pre-5.2 Data Conversion environment.*

UNIX and z/OS

1.   Edit FTP job for the correct server name or IP Address
Script used: jftp

Before running this job, you must edit it to use the correct server name or IP Address of the machine where The Solution Series is installed. Open the job in a text editor and add the server name or IP Address to the following line:

SET FTPTOSYS=

2.   Edit FTP job for the correct platform
Script used: jftp

Before running this job, you must edit it to use the proper ftp commands script. Open the job in text editor and add one of the following filenames, depending on which platform you are installing:

z/OS filename: ftpcmds_ds.os2

z/OS relational for DB2: ftpcmds_db2_ds.os2

UNIX filename: ftpcmds_ds.unx

Add the correct ftp command filename to the following line in the jftp job:

SET FTPCMDS=

Save the changes once complete.

3.   Transfer the files
Script used: jftp

Change the following to point to proper File System or Data Area or High Level Qualifier within the 5.2 system. Because the FTP commands are located at [sol series dir]\Distrib, you must edit the following FTP file:

Platform	FTP file	Items in FTP script to change
UNIX	ftpcmds_ds.unx	?FILESYS?
zOS Indexed	ftpcmds_ds.os2	?HLQ?
zOS DB2	ftpcmds_db2_ds.os2	?HLQ?

At the command prompt, run the edited jftp job. When entering the command to execute this job, the format should be:

```
jobname username
```

You will be prompted for the password. Review the ftpupload.log, located in the same directory as the install files, for error messages.

You should see 'Job completed'. The job will be run from the following directory:

[Solution Series directory]\Distrib

4.   Transfer the specific 5.2 conversion files
Script used: jftp

Change the following to point to proper File System or Data Area or High Level Qualifier within the 5.2 system. Because the FTP commands are located at [sol series dir]\Distrib\DataConv\FTP\Scripts, you must edit the following FTP file:

Platform	FTP file	Items in FTP script to change
UNIX	ftpcmds_upd.unx	?FILESYS?
zOS Indexed	ftpcmds_upd.os2	?HLQ?
zOS DB2	ftpcmds_upd.os2	?HLQ?

At the command prompt in [sol series dir]\Distrib\DataConv\FTP\Scripts, run the edited jftp job. When entering the command to execute this job, the format should be the following:

```
jobname username
```

You will be prompted for the password. Review the ftpupload.log, located in the same directory as the install files, for error message.

You should see 'Job completed'. The job will be run from the \Scripts directory.

Task 3: Install Distributed Administration files - pre-5.2 system

1.   Extract, compile, and link Distributed Administration programs
Scripts used: JXDSPACK, JXDSTRIB

These jobs use the delivered library files (UPDDSMST for your producton environment) and P9CNVT to extract, compile, and link Distributed Administration programs (DSPACK and DSTRIB).

Refer to the delivered JCLs for any overrides that may be necessary. Machine parameters are defined in the appendix.

2.   Allocate files (z/OS only)

The following demonstrates the characteristics of FILE20:

```
 Data Set Name  . . . : CYBORG.FILE20

 General Data                      Current Allocation
  Volume serial . . . : TSO30A       Allocated cylinders : 4
  Device type . . . . : 3390         Allocated extents . : 1
  Organization  . . . : PS
  Record format . . . : VB
  Record length . . . : 640
  Block size  . . . . : 27998      Current Utilization
  1st extent cylinders: 4            Used cylinders  . . : 1
  Secondary cylinders : 5            Used extents  . . . : 1

  Creation date . . . : 2001/05/17
  Referenced date . . : 2001/05/17
```

```
  Expiration date . . : ***None***
```

The following example demonstrates the characteristics of the FILE21:

```
Data Set Name  . . . : CYBORG.FILE21

General Data                     Current Allocation
 Volume serial . . . : TSO30A      Allocated cylinders : 4
 Device type . . . . : 3390        Allocated extents . : 1
 Organization  . . . : PS
 Record format . . . : V
 Record length . . . : 850
 Block size  . . . . : 854       Current Utilization
 1st extent cylinders: 4           Used cylinders  . . : 1
 Secondary cylinders : 5           Used extents  . . . : 1

 Creation date . . . : 2001/04/25
 Referenced date . . : 2001/05/17
 Expiration date . . : ***None***
```

Phase 3: Move data into your new 5.2 environment

This phase provides detailed instructions for moving your data from your existing pre-5.2 version environment into the new 5.2 Employee Database (FILE02).

Perform the following tasks in this phase:

1. Prepare your data (pre-5.2 data conversion environment).
2. Import your data (5.2 environment).
3. Populate RDBMS tables (relational only).
4. Recreate RFT records.
5. Rebuild Alternate Keys.
6. Rebuild Phonetic and Employee ID Keys.

Task 1: Prepare your data (pre-5.2 data conversion environment)

Important!

If you are currently using Distributed Administration, be sure and perform your replications and distributions before proceeding, as the following steps configure Distributed Administration for use solely for the purpose of data conversion.

1.   Run an N99 payrun

Access the Payroll Run Process Control form (AE-SCR) and set up a pay run for all pay frequencies for all organizations.

You access this form by selecting:

Component:	🏛	Payroll Setup Processing
Process:		Payroll Processing Setup
Task:	📋	Schedule Payroll Runs

The Payroll Run Process Control form (AE-SCR) should be set up as represented in the following example:

2. **Update P20IN Batch Master File**
   Script used: jpayxtr all

   Company and employee data for all organizations will be extracted from the online Employee Database and placed in FILE12.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpayxtr**
UNIX	$[CyborgHome]/runs/**jpayxtr**
z/OS	?HLQ?.JCL.CNTL(**JPAYXTR**)

3. **Perform a pay run**
   Script used: jpayrun

   Note: *If you use Interactive Workforce and have modified the jesspyrn jobstream, run it instead of jpayrun.*

   Review the P05RDR file to make sure a 'Y' is in position 18 to ensure the input of the recycle file. The P2EDIT, P4CALC, and P5PRNT programs will be processed.

   Include either an S (P05T81 only) or a B (P05T80 and P05T81) in column 19 in the P05RDR file.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpayrun**
UNIX	$[CyborgHome]/runs/**jpayrun**
z/OS	?HLQ?.JCL.CNTL(**JPAYRUN**)

Check your output listings for any anomalies.

4. Perform a maintenance run to create pay history
Script used: jmntrun

To create pay history and labor records, and apply check numbers to the newly created history records on the P20IN Batch Master File, execute the JMNTRUN jobstream. The P2EDIT, P4CALC, and P5PRNT programs will be processed. The P20 output from this job will be input into the JCONVP20 process.

Note: If you modified position 19 in the PO5RDR card, you should remove it from position 19 now.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jmntrun**
UNIX	$[CyborgHome]/runs/**jmntrun**
z/OS	?HLQ?.JCL.CNTL(**JMNTRUN**)

5. Obtain a generators-only P20
Script used: jp20strt (US)
                jp20strtc (Canada)

Edit the job so that so that the P20IN.MNT reference is changed to P20IN.GEN. the P20IN.GEN is used as one of the two P20 files used in the JDSPACK process.

Input files:

    P05T80   (P9STRT)
    P05T81   (P9CBSV for US or P9CBSVC for Canada)

The P2EDIT, P4CALC, and P5PRNT programs are processed.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jp20strt**
UNIX	$[CyborgHome]/runs/**jp20strt**
z/OS	?HLQ?.JCL.CNTL(**JP20STRT**)

6. Apply the data conversion programs to your System Control Repository (FILE01)
Script used: jmainti

Use the MAINTI program to apply the CONVERT.05 and CONVERT.04 as input files, containing the FIXP20,CONLZC, CV02PT and CRTUDS CSL programs to your current pre-5.2 version system.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jmainti**
UNIX	$[CyborgHome]/runs/**jmainti**
z/OS	?HLQ?.JCL.CNTL(**JMAINTI**)

7. Performing a reload
Script used: jreload

Recompiles Cyborg Scripting Language programs in the System Control Repository.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jreload**
UNIX	$[CyborgHome]/runs/**jreload**
z/OS	?HLQ?.JCL.CNTL(**JRELOAD**)

Check your audit messages. Reconcile any errors before continuing the upgrade process.

8. Extract, compile, and link the CONVERT program
Script used: jconvert

Use CONVERT and CBSVB to extract, compile, and link the CONVERT program.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jconvert**
UNIX	$[CyborgHome]/runs/**jconvert**
z/OS	?HLQ?.JCL.CNTL(**JCONVERT**)

9. Extract and convert Employee Database records
Script used: jconv02

Execute jconv02 for LZC to extract, convert and apply the FILE02 Employee Database records.

Warning!

JCONV02 can only be executed once.

Platform	Location and script
Windows	[Solution Series directory]\Runs**jconv02**
UNIX	$[CyborgHome]/runs/**jconv02**
z/OS	?HLQ?.JCL.CNTL(**JCONV02**)

10. Turn on Distributed Administration
Script used: JDSRSET

The JDSRSET job turns on Distributed Administration, making your pre-5.2 data conversion system ready to create the files used to apply converted data to the 5.2 system.

11. Create the data dictionary
Script used: jconvp20

FIXP20 extracts field information from FILE01and creates a data dictionary. CRUTDS configures your system for the conversion process. CONVERT is then run, with the FILE10 output from FIXP20 and an up-to-date full P20 as input. A new P20 is created. The input P20 and output P20 should be the same size; sizes may be different if invalid records are excluded from the converted file. The data will be 'cleaned up', but corrupted data will cause errors. Address data problems due to corrupted records by recreating the data P20.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jconvp20**
UNIX	$[CyborgHome]/runs/**jconvp20**
z/OS	?HLQ?.JCL.CNTL(**JCONVP20**)

12. Compare P20 files and log the differences
Script used: jdspack

Run DSPACK, with two P20s as input (one that includes the report generators, and one should be the P20 created as output from the JCONVP20 process). A FILE20 is populated with your data.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jdspack**
UNIX	$[CyborgHome]/runs/**jdspack**
z/OS	?HLQ?.JCL.CNTL(**JDSPACK**)

13. Access the Distribution Access Log Table form (DSRULE)
Access this form by making the following selections from the Navigator:

Component:		Distributed Administration
Process:		Distribution Rules/Log Table
Task:		Distribution Access Log Table

14. Identify the target DL
Enter the 5-position alphanumeric Node ID of the target DL.

15. Enter a valid organization ID associated with the source DL
Enter a 6-position organization identifier. By entering the ID here, you are indicating that the associated data from the source node may be distributed to this target DL.

Note: *If you plan on distributing option lists and/or tables from the System Control Repository, a 'ZFILE1' organization must be configured on the Distribution Access Log Table form (DSRULE) for the target DL that will be given access or you will receive a FILE01 error.*

16. Enter date to initiate distribution for this DL

Enter a date in the format MM-DD-CCYY or CCYYMMDD. This date may be the current date, a previous date, or a future date that tells the system to begin distributing the data.

Note: *It is suggested that you enter a date that has already passed, such as 19250102 (January 2, 1925).*

The form should look similar to the following example:

17. Click Save or press Enter

The Date text box changes to the format MM-DD-CCYY and the Time text box is populted by zeros. This form displays the message ----New table entry has been established----. Data distribution for this target DL is initialized.

18. Repeat steps 15 through 17 for every organization
19. Repeat steps 13 through 18 for each target node
20. Distribute the FILE20 to FILE21
Script used: jdstrib

Run DSTRIB with FILE20 as input. A new FILE21 is created and populated in a format transferable to your new 5.2 system.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jdstrib**
UNIX	$[CyborgHome]/runs/**jdstrib**
z/OS	?HLQ?.JCL.CNTL(**JDSTRIB**)

Task 2: Import your data (5.2 environment)

1. Accept the FILE21 from your pre-5.2 version system into the 5.2 system
Script used: jdsrecv

Place a 'Y' in column 31 of the FILE04 reader card. Failing to set this flag will result in an 'Incorrect segment length' error in the first record with the 'conv' node id.

Run DSRECV with FILE21 (from your pre-5.2 version system) as input. A new FILE20 is created.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jdsrecv**
UNIX	$[CyborgHome]/runs/**jdsrecv**
z/OS	?HLQ?.JCL.CNTL(**JDSRECV**)

2. Apply your pre-5.2 version data to the 5.2 system
Script used: jdsaply

Run DSAPLY, with the new FILE20 as input.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jdsaply**
UNIX	$[CyborgHome]/runs/**jdsaply**
z/OS	?HLQ?.JCL.CNTL(**JDSAPLY**)

Note: *If you are working in a relational environment, you may receive 'SQL Limit Reached - Commit Performed' messages in the log file. Disregard these messages—there is no error.*

3. Turn off Distributed Administration
Script used: jduset

The JDSUSET job turns off Distributed Administration.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jduset**
UNIX	$[CyborgHome]/runs/**jduset**
z/OS	?HLQ?.JCL.CNTL(**JDUSET**)

4. Apply converted Employee Database data
Script used: jloadhr

To apply converted Employee Database information, run JLOADHR. Use the output from Step 9 in the previous Task (1) as the FILE05 input.

Platform	Location and script
Windows	[Solution Series directory]\Runs**jloadhr**
UNIX	$[CyborgHome]/runs/**jloadhr**
z/OS	?HLQ?.JCL.CNTL(**JLOADHR**)

Check FILE03 to determine if there were any errors. A 'COMMIT FORCED BY SQL LIMIT' error may occur, but you can ignore it.

> ***You have now moved your data from your pre-5.2 version system to the new 5.2 system.***

Task 3: Populate RDBMS tables (relational only)

Script used: jpopf01

To synchronize and populate the System Control Repository and relational tables for option lists and specific application tables (Cx through Xx), execute the JPOPF01 script.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jpopf01**
UNIX	$[CyborgHome]/runs/**jpopf01**
z/OS	?HLQ?.JCL.CNTL(**JPOPF01**)

Task 4: Recreating RFT records

Script used: jf-xref/JFXREF

Run the jf-xref/JFXREF program to build and maintain the Field Name Table cross-reference menu.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jf-xref**
UNIX	$[CyborgHome]/runs/**jf-xref**
z/OS	?HLQ?.JCL.CNTL.(**JFXREF**)

Task 5: Rebuild Alternate Keys

Script used: jbldaky

Run this script to update the alternate keys built during the initial installation.

Platform	Location and Script
Windows	[Solution Series directory]\Runs**jbldaky**
UNIX	$[CyborgHome]/runs/**jbldaky**
z/OS	?HLQ?.JCL.CNTL(**JBLDAKY**)

Task 6: Rebuild Phonetic and Employee ID Keys

1.   **Indicate the type of key to build**

You do this on the second panel of the Organization Options (AF-SCR) form.
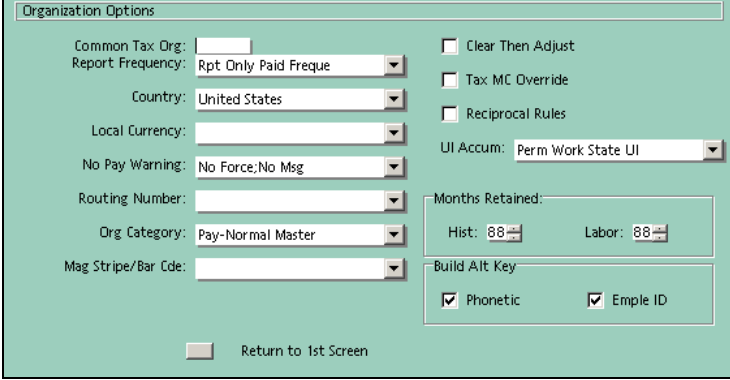
You access this form by selecting:

Component: Payroll Setup Processing
Process: Organization Setup
Task: Organization Options

and then selecting the 'More Options' button.

2. Select the type of key(s) to build

You can choose Phonetic and/or Employee ID. The indicated keys will be built automatically for new or transferred employees.

3. Save the form

4. Delete Phonetic/Employee ID Keys

The first step in rebuilding the Phonetic and/or Employee ID Keys is to delete them. Generally, you would do this after loading production data during an upgrade.

To delete all phonetic keys, perform the following steps:

1. Run the Delete All Phonetic Keys (DEL-PE) program.
   To run this program from the Navigator, select:

   Component: Development Tools
   Process: System Control Repository Utilities
   Task: Delete All Phonetic Keys

2. Click OK or press Enter

All Phonetic and Employee ID Key records are deleted for all organizations.

5. Rebuild Phonetic/Employee ID Keys from the System Control Repository

The KEY-PE program requires a two-line control record. The control records for each Phonetic Key type and each Organization Control Number value must be added to the KEY-PE job. Each control record must have the appropriate FROM and TO values.

The following are the input files, output files, and the program you need to execute KEY-PE.

To execute this utility, you need to run CBSVB as follows:

INPUT	FILE01	System Control Repository
	FILE02	Employee Database
	FILE04	Control Record File
OUTPUT	FILE03	Audit/Message File
EXECUTE	CBSVB	

For control record line 1:

In these positions	Enter
17–22	Organization Control Number value
23–28	QUERY
56–61	KEY-PE
62–63	00
64–74	FROM Employee Number (positions 1–11)
75	Continuation character '*'

For control record line 2:

In These Positions	Enter
16–23	FROM field entry (positions 12–19)
26–44	TO Employee Number

Control record example:
Phonetic and/or Employee ID Key Rebuild for organization 999999:

```
          1    1    2    2    3    3    4    4    5    5    6    6    7    7    8
....5....0....5....0....5....0....5....0....5....0....5....0....5....0....5....0
P QUERY LINE 1  999999QUERY                             KEY-PE000           *
P QUERY LINE 2          999999999
```

Phonetic and/or Employee ID Key Rebuild for organization 995555:

```
          1    1    2    2    3    3    4    4    5    5    6    6    7    7    8
....5....0....5....0....5....0....5....0....5....0....5....0....5....0....5....0
P QUERY LINE 1  995555QUERY                             KEY-PE000           *
P QUERY LINE 2          999999999
```

Phase 4: Initialize updates to custom WRITER programs

Revisions have been made to the underlying system, but before proceeding, any custom programs in your system that were created using Solution View (WRITER) must be initialized.

An easy way to identify a WRITER program is that (so long as your naming conventions adhere to standards), the program ID begins with either a '~' or an 'X'.
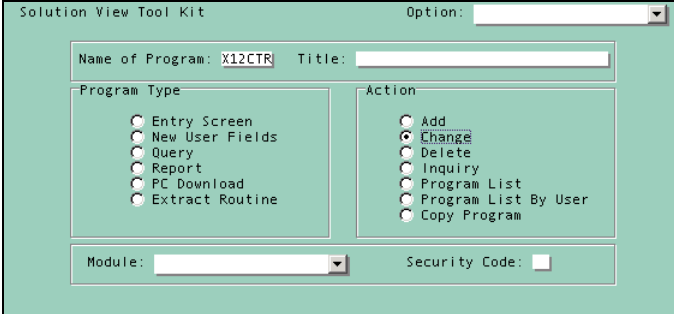
1.   Launch Solution View (WRITER)

Component:	🔧	User Tools
Process:		User Tools
Task:	🔲	Solution View

2.   Type the program ID of the custom program in the Name of Program field

3.   Select the Change radio button option in the Action area

The screen should look similar to the following:

4.   Press Enter

Repeat this procedure for all your custom programs. This procedure initializes the custom programs against the new version of The Solution Series.

Phase 5: Analyze garnishments

If you choose to convert to the new Garnishments model, you will need to perform some analysis of your existing Garnishments configuration.

Four spreadsheets are available to use as part of the conversion process—one spreadsheet per new Garnishments form (Garnishments Administration 1 through 4 (GR1SCR, GR2SCR, GR3SCR, GR4SCR)). Complete each of these spreadsheets with the relevant information and transfer the information into The Solution Series using the Import feature.

☞ *Refer to the Optimizing System Features documentation for detailed information on using the Import functionality.*

Phase 6: Install and configure the administrative client

Perform the following tasks to install and configure the Administrative Client:

1. Download the administrative client update.
2. Run the update executable.

Note: After running any processes, review the output to determine if there were any errors.

1. Download the administrative client update.
2. Run the update executables.

Task 1: Download the administrative client update

1.   Create a directory on a PC to store the downloaded files

2.   Access CUBBS

☞   *Refer to Accessing CUBBS in the knowledgebase or Technical Administration guide for detailed instructions for logging onto CUBBS.*

3.   Go to the 5.2 page

Scroll through the page and click on the appropriate links to download the installation program to your download directory.

To configure the replacement for the administrative client, refer to the *Installing and Configuring The Solution Series 5.2* guide for your platform for detailed information and instructions on performing the necessary tasks. You can download this document in PDF format from the Customer Center.

Important! The AutoInstall installs a new client or updates an existing pre-5.2 client with the 5.2 files. If you install a new client without deleting the FILECL32, you may receive errors when you try to connect to the existing environment.

Task 2: Run the update executable

1.   Run the update executable

Start the update process by clicking the Install_admin_client_v52 executable from the folder to which it was downloaded. Follow the InstallShield instructions.

2.   Automatic File Backup

Your 5.1 csss32.exe, MergeLetter.dll, and CybCfg.exe will be backed up in a folder called 'backup' located in the current Client51 directory.

3.   Indicate if you are running on a z/OS system

4.   Check your Setup Information

At this point, you can click Back to review your setup data or click Next to continue.

5.   Launch Connection Editor

At this point in the upgrade, the Connection Editor launches automatically.

To configure the5.2 replacement for the administrative client, refer to the *Installing and Configuring The Solution Series 5.2* guide for your platform and use the existing instructions. You can download this document in PDF format from the Customer Center.

Phase 7: Configure security

The tasks in this phase need to be carried out by your Security Officer:

1. Add Security Records.
2. Implement Security Enhancements.

Task 1: Add Security Records

There are two security records which need to be added by your Security Officer. It is not possible to use MAINTI to perform this operation.

Please have your Security Officer refer to the *Setting Up and Maintaining Security* guide for further details. The relevant information can be found in the appendix Adding Secure Records During Installation/Upgrade.

Task 2: Implement Security Enhancements

There are new security features with 5.2 which have been added to comply with the tightened security requirements imposed by the Sarbanes Oxley Act.

To implement these features, your Security Officer should refer to the *Setting Up and Maintaining Security* guide for details about the enhancements and for instructions on installing them. The relevant section can be found in the appendix Activating and using Extended Security.

PART 3

Appendices

In This Section

A P P E N D I X   A

Expand Transactions

In This Appendix

Introduction

Due to an increasing amount of Company and Employee data being stored on the client side, Hewitt has significantly increased the size of several storage areas in Solution Series 5.2:

P4CALC/O4CALC

- Company area – PAYER
- Employee area – PAYEE

CBSV

- AREA2 – employee
- AREA3
- AREA4 – company header

EXPAND area increases

We have significantly enlarged the sizes of the Company (PAYER), Employee (PAYEE), and Tax area.

P4CALC/O4CALC

In P4CALC/O4CALC, the PAYER and PAYEE areas have been increased by 100,000 characters each. Also, the TAX AREA has increased:

EXPAND Area	New area sizes for 5.2 (bytes)
Company area (PAYER)	132,271
Employee area (PAYEE)	124,788
TAX AREA	784,000

Note: *You have the option to reduce the new size of the TAX AREA. If a "T" is added to the Machine Parameter string, then 720,000 bytes will be removed leaving 64,000.*

CBSV

In CBSV, AREA 2 and AREA 4 have been increased by 100,000 characters each:

EXPAND Area	New area sizes for 5.2 (bytes)
AREA 2 (employee)	129,158
AREA 3	232,767
AREA 4 (company)	132,293

Important!

The pre-5.2 EXPANDs will not function in extracting programs from the 5.2 CYBMST. You will need to remove or modify your EXPAND control records before upgrading to 5.2.

Requirements for the 5.2 upgrade

Hewitt has significantly changed the layout of the Batch System expand records. Your pre-5.2 expand transactions will not function when extracting programs from CYBMST. Based on the new values listed above, you may be able to remove the following expansions:

- EXPAND TAX
- EXPAND PAYER
- EXPAND EMPLOYEE

Note: *During this process, you should monitor the Payroll Audit Trail (0101) line 'POSITIONS LEFT IN DATA AREA'.*

If you decide to keep any expand records, be aware that the EXPAND value has moved from columns 25/29 to 25/31 with leading zeros:

```
        1         2         3
1235678901234567890123456789012345
EXPAND TAX           0005000
EXPAND REPORT BATCH    0010000
```

If you wish to use the H2 'A' option to load all the U.S. tax tables, an EXPAND of 200,000 is recommended (place a 'A' in position 27 of the H2 parameter transaction - P05RDR file for P4CALC step):

```
        1         2         3
1234567890123456789012345678901234512345
EXPAND TAX             0200000
```

In The Solution Series application, the 'Expand Areas in CBSV Programs' form controls the expansion of AREA2, AREA3, and AREA4 – these expansions can now be set to zero.

Note: *Do NOT reduce the DATABASE COMMIT value.*

You can run the RECSIZ CSL program to determine the largest Employee and Company records in your FILE02 and how they relate to the new area sizes in CBSV.

Aᴘᴘᴇɴᴅɪx B

Component Lists

In This Appendix

Server Data

Common

p05rdrqt.dat

file4mnt

file5mnt

re5204

re5205

re52m04

re52m05

cvbmst

cbsv

taxfile

taxfilec

Windows

JCONV02_REPORT.bat

JLOADHR.bat

Enhanced Reporting / DDI

MMLRecs_EnhancedRpt.dat

MMNP2Recs_EnhancedRpt.dat

UNIX

Indexed

cbsv.ovr

eprddi05

p9cbsv.04

p9cbsvc.04

Demo0105_4-byte_ASCII

Windows

Indexed

Cbsv.ovr

eprddi05

Islock.exe

O4PRT1NR

P4PART1

P9CBSV.04

JCONV02.bat

JLOADHR.bat

Demo0105_4-byte_ASCII

Oracle

f1rstr.04

f2rstr.04

O4PART1

SQL

F1RSTR.04

F2RSTR.04

O4PART1

z/OS

Demo0105_4-byte_EBCDIC

Indexed

P9CBSC04

P9CBSV04

P9STRT04

SBMidx05

DB2

P9CBSV04

P9STRT04

SBMdb205

Java Related

jbldcab

Jbldgfac

Server Programs

Common

CBSVRFT.CBL

UNIX

Indexed

cbsvb.cob

cbsvb.mf2

bsvbt.cob

cbsvo.cob

cbsvot.cob

cybio.cob

p20cnvt.cbl

p45sort.cob

p9cnvt.cob

repsort.cob

sha.cob

AIX

cybsha1.o

HP

cybsha1.o

Solaris

cybsha1.o

Oracle

RDBPGM0.cob

cbsvbr

cbsvb.pco

Windows

Indexed

CBSVB.CBL

cbsvb.mf2

CBSVBT.CBL

CBSVO.CBL

CBSVOT.CBL

CybIO.exe

cybsha1.obj

p20cnvt.cbl

p45sort.cbl

P9CNVT.CBL

repsort.cbl

sha.cbl

Oracle

CBSVB.pco

CBSVBR

RDBPGM0.cbl

SQL

CBSVB.CBL

CBSVBR

RDBPGM0.CBL

z/OS

Indexed

CBSVADV

CBSVB

CBSVBOL

CBSVBOS2

CBSVBT

CBSVO

CBSVOT

CBSVSETF

CSTCFG

CYBIO

EXCICBSV

EXCIREPL

EXECBSV

CYBSHA1

CICS

CAB.jar

CABSrv.jar

CLRTSQS.CBL

dfjjvmpr.props

ftpcmds_cab.os2

ftpcmds_ior.os2

jftp_CAB.bat

DB2

cbsvb

cbsvbr

RDBPGM0

Server Scripts

UNIX

EPR

BESS

BISLOCK

BISW

BIWE

BPAY

BXTR

P247PAY

RSPAWNPAY

U247UPD

Installation

jinstallst

Indexed

j52up1

j52up2

j52upm

jcmpcvbn

jcmpcvn

jcmpp20

jcmprft

jcmpsort

jconvp20

jcrtp20

jcrtp20c

jcrossx

jdemo023

jdocxtr

jdscr08

jiswasx

jloadessgen

jloadgen

jlogexp

jlogext

jmntf08

jmntrun

jmtoanl

jmtoclr

jmtolod

jp20cnvt

jp20strc

jp20strt

jpaymrg

jpaymrg_full

jpayrun

jpayxtr

jqtrrun

jupdateU

jupdgen

Oracle

jcf1rstr

jcf2rstr

jf1rstr

jf2rstr

AIX

jcmpcvbr

jcmpcvr

jxo4calr

HP Solaris

jcmpcvbr

jcmpcvr

jxo4calr

Windows

EPR

bess.bat

BHLD.BAT

BHRD.BAT

Bislock.bat

BISW.BAT

BIWE.BAT

Bmrg.bat

Bpay.bat

Brpt.bat

BXTR.BAT

P247LAUNCH.BAT

P247PAY.bat

RspawnPAY.BAT

RSXLAUNCH.bat

RSXUPD.BAT

U247LAUNCH.BAT

U247UPD.bat

Indexed

j52up1.bat

j52up2.bat

j52upm.bat

jclean01.bat

jcmpcbio.bat

JCMPCVBN.BAT

jcmpcvn.bat

JCMPCVON.BAT

Jcmpp20.bat

JCMPRFT.BAT

jcmpsort.bat

jcompcvb.bat

jcompcvo.bat

jcrossx.bat

jcrtp20

jcrtp20c

jdemo023.bat

jdocxtr.bat

jdscr08.bat

jessdemo.bat

jfullmrg.bat

JISWASE.BAT

jloadESSgen.bat

jloadgen.bat

Jlogexp.bat

Jlogext.bat

JMNTF08.BAT

jmntrun.bat

jmtoanl.bat

jmtoclr.bat

jmtolod.bat

JP20CNVT.BAT

jp20strc.bat

jp20strt.bat

jpaymrg.bat

jpayrun.bat

jpayxtr.bat

jqtrrun.bat

JUpdateW.bat

jupdgen.bat

jxcybmst.bat

mfextfh.cfg

MFSETUP.BAT

Oracle

jcf1rstr.bat

jcf2rstr.bat

jcmpcvbr.bat

jcmpcvr.bat

jcmprdb0.bat

jf1rstr.bat

jf2rstr.bat

jpopf01.bat

jxo4calr.bat

jcmpsha1.bat

SQL
jcf1rstr.bat

jcf2rstr.bat

jcmpcvbr.bat

jcmpcvor.bat

JCMPCVR.BAT

jf1rstr.bat

jf2rstr.bat

Submit View
jqry.bat

jrpt.bat

z/OS

DB2
DATAPROC

J52UP1

J52UP2

J52UPM

JASSEMBL

JBACKEM

JBINDCYB

JBINDF

JBLDAKY

Jbldcfg

JCBSVB

JCBSVBN

JCBSVBT

JCBSVO

JCBSVOT

JCF1RSTR

JCF2RSTR

jclean01

jclean3x

JCMPCBIO

JCMPCFG

JCMPCVBR

JCMPF1RS

JCMPP9C

jcmppgms

JCMPRDB0

JCMPRDBH

JCMPSUBR

JCOMCBSV

JCOMDB2O

JCOMPRFT

JCONVERT

JCONVERT.OS

JCONVNA

JCONVP20

JCONVP20.OS

JCROSSX

JCRTCYB

JCRTPGMS

Jcvtcfg

Jcvtrstr

JDATASET

JDEFINE

JdefSS

Jdefv22

JDEMO01

JDSCR08

JEMPDATA

JESSiswe

JESSldgn

JESSmtrn

JESSpmrg

JESSpxtr

JESSpyrn

JESSxpt

JESSxrg

JEXPORT

jexportr

JF1RSTR

JF2RSTR

JFULLMRG

Jfxref

jhrdemc

JHRDEMO

JISWASX

JLOGEXP

JLOGEXT

JMAINTI

JMAINTO

JMAKECL

JMNTF08

JMNTRUN

JMTOANL

JMTOCLR

JMTOLOD

JP20STRT

JPAYMRG

JPAYRUN

JPAYXTR

JPOPF01

jprdemc

JPRDEMO

JPULCVBR

JPULLRDB

JREBUILD

JRELOAD

JREPORT

Jtooldat

jupdateo

JUPDCYBM

JXCYBMST

JXO4CALC

JXP2EDIT

JXP4CALC

JXP5PRNT

jxp9cbsc

JXP9CBSV

JXP9CNVT

JXP9STRT

JXREPT20

JXRPTGEN

JXRPTQTR

JXSUBRTN

Indexed

DATAPROC

J52UP1

J52UP2

J52UPM

JASSEMBL

JBACKEM

JBLDAKY

Jbldcfg

JCBSVB

JCBSVBT

JCBSVO

JCBSVOT

jclean01

jclean3x

JCMPCBIO

JCMPCFG

JCMPCVBN

JCMPP9C

JCMPPGMS

JCOMCICS

JCOMPBOL

JCOMPRFT

JCONVERT

JCONVERT.OS

JCONVNA

JCONVP20

JCONVP20.OS

JCRTP20

JCRTP20C

JCRAUDIT

Jcvtcfg

Jcvtrstr

JDATASET

JDEFBCYL

JDEFBOL

JDEFCFG

Jdefgrp

JDEFINE

Jdefomvs

JdefSS

Jdefv22

JDELETE

JDEMO01

jdemo023

JDSCR08

JEMPDATA

JESSiswe

JESSldgn

JESSmtrn

JESSpmrg

JESSpxtr

JESSpyrn

JESSxpt

JESSxrg

JEXPORT

JFULLMRG

Jfxref

jhrdemc

JHRDEMO

JISWASX

JLOADGEN

jlogexp

jlogext

JMAINTI

JMAINTO

JMAKECL

JMNTF08

JMNTRUN

JMTOANL

JMTOCLR

JMTOLOD

JP20STRC

JP20STRT

JPAYMRG

JPAYRUN

JPAYXTR

jprdemc

JPRDEMO

JPULCVN

JQTRRUN

JRAUDIT

Jrcvpdse

JRDOBOL

JREBUILD

JRELOAD

JREPORT

JRESUME

JRSXFTP

JSUSPEND

Jtooldat

jupdateo

JUPDCYBM

JUPDGEN

JXCYBMST

JXO4CALC

JXP2EDIT

JXP4CALC

JXP5PRNT

jxp5qtr

JXP7COMP

jxp9cbsc

JXP9CBSV

JXP9CNVT

JXP9STRT

JXREPT20

JXRPTGEN

JXRPTQTR

JXSUBRTN

Server Events

Common

CyborgEventService.exe

CybEvtSvc.cpl

**DLLs
Registered**
BackendEventsLibrary.dll

WebApp

srvevent.war

Canada
srvevent_fr.war

Tax Files

U.S.

taxfile

Canada

Taxfilec

A P P E N D I X   C

Report Generators

In This Appendix

5.2 Report Generators

The following table lists the Report Generators (RGs) delivered with 5.2:

Generator	Name as listed on CYBMST	Generator Name	New/Changed
010A	R.RPT0A	PERMANENT CONSTANTS	
010G	R.RPT0G	VARIABLE CONSTANTS V1.0	
010O	R.RPT0O	WRKFLDS	
010P	R.RPT0P	PAY CONSTANTS	
0100	R.RPT00	ERRORS AND WARNINGS V1.1	
0100	R.RPT00F	ERREURS ET AVERTIS.1.0F	
0101	R.RPT01	PAYROLL AUDIT TRAIL V1.0	
0101	R.RPT01F	VÉRIFICATION - PAIE V1.1	
0103	R.RPT03	CONTROL HEADERS V3.01	Changed
0103	R.RPT03F	EN-TÊTES CONTRÔLE V2.00F	
0117	R.RPT17	EDIT ERROR MESSAGES 1.0	
0118	R.RPT18	ROUTINE NUMBERS 000-099	
0119	R.RPT19	ROUTINE NUMBERS 100-255	
0120	R.RPT20	DEFINE EDIT TABLES V1.0	Changed
0121	R.RPT21	FIELD NUMBERS 1-100	
0122	R.RPT22	FIELD NUMS 101-200 V1.0	
0123	R.RPT23	FIELD NUMS 201-300 V1.0	Changed
0124	R.RPT24	FIELD NUMBERS 301-400	
0125	R.RPT25	FIELD NUMBERS 401-500	
0126	R.RPT26	FIELD NUMBERS 501-600	
0127	R.RPT27	FIELD NUMBERS 601-700	
0128	R.RPT28	FIELD NUMBERS 701-800	
0129	R.RPT29	FIELD NUMS 801-900 V1.1	Changed
0202	R.RPT02	MASTER FILE PRINT V1.4	Changed
0202	R.RPT02F	DOSSIER MAÎTRE V1.01F	
0404	R.RPT04	UNLOAD MASTER DATA V1.20	Changed
0505	R.RPT05	ACCRUAL REPORT v1.01	
0505	R.RPT05F	RAPPORT ACCUMULÉ	
1C1C	R.RPT1C	PAY RECONCILIATION V1.01	
1C1C	R.RPT1CF	RAPPROCHEMENT DE PAIE 1F	
1H1H	R.RPT1H	HISTORY REPORT V1.2	Changed

Generator	Name as listed on CYBMST	Generator Name	New/Changed
1H1H	R.RPT1HF	RAPPORT HISTORIQUE PAIE	
1J1J	R.RPT1J	CORRECT CITY/ST FMT V1.0	Changed
1K1K	R.RPT1K	DUMMY GEN FOR 1J1J ADJS	
1L1L	R.RPT1L	LABOR REPORT 40.0	
1L1L	R.RPT1LC	LABOUR REPORT 40.0	
1L1L	R.RPT1LF	RAPPORT MAIN-D'OEUVRE 40	
1M1M	R.RPT1M	Flat Rate Tax Filing Rpt	
1S1S	R.RPT1S	SSA EVS INTERFACE V1.10	
1T1T	R.RPT1T	R1-SRC/R2-SCR RPT V1.00	
1Y1Y	R.RPT1Y	DELETE WLFDW2 DATA	
1Z1Z	R.RPT1Z	TAX ADJ FROM MEMO HEDS.	
2B2B	R.RPT2B	FLAG ACTIVE J'S v1.0	Changed
2C2C	R.RPT2C	OUTSTANDING RECON NBRS.	
2C2C	R.RPT2CF	NOS RAPPRO. EN CIRCUL.	
2F2F	R.RPT2F	0 FREQS.-TERM. EMPS. 2.0	Changed
2F2F	R.RPT2FF	FRÉQ.ZÉRO-CESSATION EMP	
2G2G	R.RPT2G	0 GARNS.-TERM. EMPS. 1.0	New
2H2H	R.RPT2H	HED'S-COMB REG LARGE	Changed
2H2H	R.RPT2HF	HED'S-COMB REG LARGE	Changed
2K2K	R.RPT2K	GARNISHMENTS ACTIONED	New
2L2L	R.RPT2L	PAYROLL SUMMARY REP V1.0	New
2M2M	R.RPT2M	MEMO HED'S	Changed
2M2M	R.RPT2M-COMREG	MEMO HED'S + COM REG	New
2M2M	R.RPT2MF	MÉMO REGISTRE HED GROSS	Changed
2M2M	R.RPT2MF+COMREG	MÉMO REGISTRE + COM REG	New
2P2P	R.RPT2P	PHILA TAX REPORT	
2R2R	R.RPT2R	PITTSBURGH TAX REPORT	
2S2S	R.RPT2S	WAGE SUPPLEMNTL RPT V1.1	Changed
2T2T	R.RPT2T	TAXES-COMB REGISTER V1.3	Changed
2T2T	R.RPT2TF	IMPÔTS-REGISTRE COM 1.0F	Changed
2U2U	R.RPT2U	FUI WAGES BY STATE	
2W2W	R.RPT2W	AUDIT TRAIL WARNINGS	
2W2W	R.RPT2WF	AVERTIS. VÉRIFI.PAIE	
2X2X	R.RPT2X	OUT OF BALANCE CHECK	
2X2X	R.RPT2XF	VÉRIF.HORS ÉQUILIBRE	
2222	R.RPT22	COMBINED REGISTER	Changed

Generator	Name as listed on CYBMST	Generator Name	New/Changed
2222	R.RPT22F	REGISTRE COMBINÉ	Changed
3U3U	R.RPT3U	Tax Arrears Report V1.0	New
4C4C	R.RPT4C	X CARDS-MANUAL RECON	
4L4L	R.RPT4L	SUMMARIZE LABOR V1.0	Changed
4L4L	R.RPT4LF	SOMMAIRE MAIN-D'OEUVRE	
4R4R	R.RPT4R	ER CARDS-MANUAL RECON	
4S4S	R.RPT4S	SUPPLEMENTAL WGS RPT 1.0	New
4W4W	R.RPT4W	NON-SUBMITTED W-4'S	
4X4X	R.RPT4X	MODIFY WITHHOLDING	
4040	R.RPT40	CREATE ER VOID(CARD)	
4141	R.RPT41	TRANSFER BASIC DATA V1.0	
4242	R.RPT42	TRANSFER AMOUNTS	
4245	R.RPT45	PAYROLL ER VOID(RECYCLE)	
4343	R.RPT43	HISTORY REVERSAL	
4646	R.RPT46	FOR. CURR. SCHED. V1.0	
4747	R.RPT47	COST CENTRE SUMM V1.0	
4848	R.RPT48	FOR. CURR. BY EMP V1.0	
4949	R.RPT49	ALL CURR BY EMP V1.0	
5G!A	R.RPT!A	CONSUMER ALASKA V1.0	New
5G!B	R.RPT!B	CONSUMER ARKANSAS V1.0	New
5G!C	R.RPT!C	MC !C FOR COLUMBIA V1.0	New
5G!D	R.RPT!D	MC !D FOR HAWAII V1.0	New
5G!E	R.RPT!E	MC !E FOR MO NE V1.0	New
5G!F	R.RPT!F	MC !F FOR NJ AND NY	New
5G!G	R.RPT!G	MC !G FOR ND SD TN V1.0	New
5G!H	R.RPT!H	MC !H FOR OREGON V1.0	New
5G!I	R.RPT!I	CONSUMER GARN RI V1.0	New
5G!J	R.RPT!J	CONSUMER GARN TN SD V1.0	New
5G!K	R.RPT!K	CONSUMER GARN MD OR 1.0	New
5G!L	R.RPT!L	CHILD SUPP CA V1.0	New
5G!M	R.RPT!M	CHILD SUPPCT V1.0	New
5G!N	R.RPT!N	CHILD SUPP NC V1.0	New
5G!O	R.RPT!O	MC !O FOR OREGON V1.0	New
5G!P	R.RPT!P	MC !P FOR RI V1.0	New
5G!Q	R.RPT!Q	WAGE ASSIGN ARIZONA V1.0	New
5G!R	R.RPT!R	NM CA MT SD VT VA V1.0	New

Generator	Name as listed on CYBMST	Generator Name	New/Changed
5G!S	R.RPT!S	WAGE ASSIGN COL. V1.0	New
5G!T	R.RPT!T	WAGE ASSIGN ILL. V1.0	New
5G!U	R.RPT!U	WAGE ASSIGN MASS V1.0	New
5G!V	R.RPT!V	WAGE ASSIGN NEW YORK 1.0	New
5G!W	R.RPT!W	WAGE ASSIGN RI WI V1.0	New
5G!X	R.RPT!X	WAGE ASSIGN W.VIR V1.0	New
5G!Y	R.RPT!Y	WAGE ASSIGN AZ WV V1.0	New
5G!Z	R.RPT!Z	MC !Z FOR WV V1.0	New
5G!1	R.RPT!1	WAGE ASSIGN !1 V1.0	New
5G+C	R.RPT+C	FEDERAL TAX LEVY	New
5G+D	R.RPT+D	SUPPORT LEVY	New
5G+E	R.RPT+E	OTHER LEVY	New
5GPC	R.RPTPC	GST M.C.	
5G1B	R.RPT1B	NAME FORMAT SUBRTN v003	
5G2L	R.RPT2L	2L2L SUBROUTINE V1.0	New
5G35	R.RPT35	BENEFITS ROUTINES	
5G4F	R.RPT4F	FRICK SUBROUTINE V1.00	
5G51	R.RPT51	FIPS POST. CODES v002	
5G6C	R.RPT6C		
5G6I	R.RPT6I	CH SUPPORT ACH TAPE V1.5	New
5G6R	R.RPT6RR	RECIPROCAL FLAG SETUP	
5G7B	R.RPT7B	RG TAX METHOD CODES V1.0	
5G7R	R.RPT7R	RECIP TAX OFFSETS V2.0	Changed
5G8A	R.RPT8A	ACCUMULATION M/C V1.04	
5G8B	R.RPT8B	MINIMUM CHECK M/C	
5G8C	R.RPT8C	INTEREST CALCULATION	
5G8G	R.RPT8G	GARNISHMENT M/C V1.03	
5G8Q	R.RPT8Q	HOURS WORKED M/C v001	
5G8R	R.RPT8R	RECIP TAX SETUP M/C V1.0	New
5G9A	R.RPT9A	WASH IND COMM 5 DEC RATE	
5G9E	R.RPT9E	DATE DRIVEN TC MC 38.0	
5G9K	R.RPT9K	ROUND NET PAY	
5G9W	R.RPT9W	HOURS WORKED M.C. V1.00	
5G9Y	R.RPT9Y	RATE X FACTOR M.C. V1.0	
5G97	R.RPT97	PC SORT BINARY	
5H5Z	R.RPT5Z	ON-LINE CSSS ROOT V1.04	Changed

Generator	Name as listed on CYBMST	Generator Name	New/Changed
5R5R	R.RPT5R	RECIP TAX OFFST RPT V1.0	New
5Z5Y	R.RPT5Y	BUILD RANDOM FILE	
5558	R.RPT58	ENDING BALANCES	
5558	R.RPT58F	SOLDE DE CLÔTURE	
5959	R.RPT59	BOND BALANCE REGISTER	
6A6A	R.RPT6A	ACH TAPE V1.4	
6B6B	R.RPT6B	CANADIAN DIRECT DEPOSIT	
6B6B	R.RPT6BF	DÉPÔT DIRECT CANADIEN	
6D6D	R.RPT6D	DIRECT DEPOSIT REG V1.2	Changed
6D6D	R.RPT6DF	REGISTRE DÉPÔT DIRECT	
6E6E	R.RPT6E	DIRDEP REG, BY PMNT DATE	
6E6E	R.RPT6EF	DÉP DIR RÉG,PAR DATE PMT	
6H6H	R.RPT6H	UPDT. L7O/LPO GARN V1.1	Changed
6I6I	R.RPT6I	ACH TAPE CH-SUP 5.2 V1.8	Changed
6K6K	R.RPT6K	DEP SLIP/COMB. REGISTER	Changed
6L6L	R.RPT6L	PAY DOC./COMB. REG. V1.2	
6R6R	R.RPT6R	CH SUPP ACH REG. V1.1	
6S6S	R.RPT6S	BILLING STATISTICS V1.0	
6262	R.RPT62	CANADIAN CHEQUE V1.20	Changed
6263	R.RPT63	CANADIAN CHEQ-FREN V1.10	Changed
6262	R.RPT62L	CDN LASER CHEQUE V1.40	Changed
6263	R.RPT63F	CANADIEN - CHÈQUE V1.10	Changed
6263	R.RPT63L	CDN CHEQUE-FREN LS V1.10	Changed
6666	R.RPT66	CAN. DEPOSIT SLIP V1.10	Changed
6666	R.RPT66F	REGIS DE DÉPÔT-CDN V1.10	Changed
6666	R.RPT66L	CAN. DEPOSIT LSR V1.10	Changed
6767	R.RPT67	DEPOSIT SLIP-COMB. REG.	
6868	R.RPT68	CHECK-COMBINED REG V1.4	Changed
7A7A	R.RPT7A	PD7A REMITTANCE V00.01	
7A7A	R.RPT7AF	VERSEMENT PD7A V00.01	
7C7C	R.RPT7C	DISTRIBUTION REPORT	
7C7C	R.RPT7CF	RAPPORT DE DISTRIBUTION	
7D7D	R.RPT7D	BLS MWR FORMT V37.01	
7E7E	R.RPT7E	Data Mart HL Xtrat V1.53	
7G7G	R.RPT7G	CAN PAYROLL SAVINGS V1.0	
7G7G	R.RPT7GF	ÉPARGNE CANADA -PAIE V1.0	

Generator	Name as listed on CYBMST	Generator Name	New/Changed
7H7H	R.RPT7H	CPS REGISTER	
7H7H	R.RPT7HF	REGISTRE OEC	
7L7L	R.RPT7L	PAY INFO EXTRACT V1.1	Changed
7L7L	R.RPT7LA	PAY INFO EXTRACT V1.1	
7L7L	R.RPT7LF	EXTRAIT INFO PAIE V1.0	
7M7M	R.RPT7M	PAYSLIP EXTRACT V1.1	Changed
7M7M	R.RPT7MA	PAYSLIP EXTRACT V1.1	
7M7M	R.RPT7MF	PAYSLIP EXTRACT V1.0	
7Q7Q	R.RPT7Q	BACK OUT QTD FIGURES	
7S7S	R.RPT7S	WCB - MONTHLY V00.01	
7S7S	R.RPT7SF	CST - MENSUEL V00.01F	
7T7T	R.RPT7T	TAX REPORTING TAPE	
7U7U	R.RPT7U	REPORT ON HIRINGS	
7V7V	R.RPT7V	CDN WCB REPORT V00.01	
7V7V	R.RPT7VF	RAPPORT CST V00.01F	
7W7W	R.RPT7W	WORKERS COMP REPORTV1.1	Changed
7Y7Y	R.RPT7Y	CDN NF WCB V01.01	
7Y7Y	R.RPT7YF	WHSCC (NF) V00.01F	
7Z7Z	R.RPT7Z	WCB-NOVA SCOTIA V00.01	
7Z7Z	R.RPT7ZF	CST-NOUVELLE ÉCOSSE 1.0F	
7575	R.RPT75	JOURNAL ENTRY-FILE V1.0	
7575	R.RPT75F	DOSSIER-ÉCRITURE JRNL1.F	
7676	R.RPT76	JOURNAL ENTRY-PRT V1.0	
7676	R.RPT76F	ÉCRITURE JOURNAL V1.00F	
7777	R.RPT77	PAYROLL ACCRU-TAPE V1.00	
7777	R.RPT77F	BANDE-PAIE ACCUMUL.V1.0F	
7878	R.RPT78	PAYROLL ACCRU-PRNT V1.00	
7878	R.RPT78F	RAP-PAIE ACCUMUL. V1.00F	
7979	R.RPT79	LABOR DISTRIBUTION V1.01	
7979	R.RPT79F	DISTR.MAIN D'OEUVRE 1.0F	
8S8S	R.RPT8S	COLLECT HOURS WORKED	
8T8T	R.RPT8T	YEAR END CLEARINGS V1.0	
8T8T	R.RPT8TF	SUPPR FIN EXERCISE V1.0	
8W8W	R.RPT8W	W-2 FORMS ESTIM. V1.0	
8Z8Z	R.RPT8Z	COLLECT HOURS WRKED V1.0	Changed
9A9A	R.RPT9A	941 SUMMARY REPORT v001	

Generator	Name as listed on CYBMST	Generator Name	New/Changed
9B9B	R.RPT9B	QUARTERLY EMP CNTS v001	
9E9E	R.RPT9E	MASTER FILE STATUS V1.10	Changed
9F9F	R.RPT9F	MASTER FIL STAT (CAN)V1.0	
9F9F	R.RPT9FF	DOSSIER MAÎTRE D'ÉTAT V1.0	
9H9H	R.RPT9H	DELETE UNUSED H & J SEGS	
9M9M	R.RPT9M	STATISTICS CANADA REPORT	
9M9S	R.RPT9S	STATS CANADA - PART 2	
9N9N	R.RPT9N	INSURABLE EARNINGS	
9R9R	R.RPT9R	RETRO PAY INCREASE	
9090	R.RPT90	PD FRQ TAX FLNG RPTV1.3	Changed
9091	R.RPT91	ALL FREQ TAX FILING V1.2	Changed
9595	R.RPT95	STATE 941A'S v788	
9$9$	R.RPT9$	FEDERAL SUMMARY RPT v001	

A P P E N D I X   D

Machine Parameters

In This Appendix

5.1 Machine Parameters

Introduction

This appendix provides detailed machine parameter information for Distributed Administration programs. These machine parameters are to be used along with the delivered library files (DSCYBMST for your 5.2 environment to extract, compile, and link Distributed Administration executables (DSTRIB, DSRECV, and DSPACK).

UNIX and Windows Platforms Using Microfocus Compiler (with or without relational database)

```
          1    1    2    2    3    3    4    4    5
1...5....0....5....0....5....0....5....0....5....0
DSTRIB    |ISEV@C           MICRO-FOCUS.
   ** C.DSTRIB
   999999
DSRECV    |ISEV@C           MICRO-FOCUS.
   ** C.DSRECV
   999999
DSPACK    |ISEV@C           MICRO-FOCUS.
   ** C.DSPACK
   999999
```

IBM Mainframe Platforms (with or without relational database)

```
          1    1    2    2    3    3    4    4    5
1...5....0....5....0....5....0....5....0....5....0
DSTRIB    OPCY             IBM-370.
   ** C.DSTRIB
   999999
DSRECV    OPCY             IBM-370.
   ** C.DSRECV
   999999
DSPACK    OPCY             IBM-370.
   ** C.DSPACK
   999999
```

A P P E N D I X E

Tables and Forms

In This Appendix

5.2 Tables and Forms

Introduction

The following tables list the tables/forms added or modified in 5.2. The information has been presented in two formats—by Solution Series form and by Table.

Solution Series forms added/changed

The following table shows the Solution Series forms that have been added or amended since 5.1:

Solution Series Forms Added/Changed since 5.1	Table	Segment
02-SCR	EMP_MILITARY_DATE	ML6F
30HSCR	FORMAL_EDUC_REQ	L6S
59CSCR	DEPENDENT_GTL_CVG	L9X
AF-SCR	PAY_PROCESS_OPT	DCAF
ARPSUB	RETROACTIVE_INFO	MLG6
AW-SCR	PR_TABLE_CTRL	YTZAW
CK-SCR	REGISTER_OVERRIDES	DCCK
FFGSCR	NAME_ADDR_REQ_NAME	L8Q
GR1SCR	GRNS_EE_ORD_PART1	L76
	GRNS_EE_ORD_PART2	L77
GR2SCR	GRNS_EE_ORD_PART3	L78
GR3SCR	GRNS_EE_ORD_PART4	L79
GR4SCR	GRNS_EE_ORD_PART5	L7O
GRASCR	GRNS_REQ_DED	YUG01
GRBSCR	GRNS_REQ_DED_HED	YUG02
GRCSCR	GRNS_DED_PRI	YUG03
GRDSCR	GRNS_HED_RANGE	YUG04
GRESCR	GRNS_TAX_LEVY_RULE	YUG05
GRFSCR	GRNS_SUPPORT_RULES	YUG06
GRGSCR	GRNS_GARN_RULES	YUG07
GRHSCR	GRNS_NDI_REDUCT	YUG08
GRKSCR	GRNS_ACH_HEAD	DCC8
	GRNS_ST_ACH_HEAD	DCC9
GRTSCR	GRNS_SWTCH_OFF	YUG13
HL-SCR	RETROACTIVE_INFO	MLG6
JTASCR	TAX_ARREARS_BLNCE	L6O

Solution Series Forms Added/Changed since 5.1	Table	Segment
OP-SCR	EMP_OVERTIME_PREM	L7D
PD-SCR	V80_BENEFIT	LPD
PEMSCR	MISC_PAY_DETAILS	L8Z
PTMSCR	PLCY_MEAL_PNLT	YT_PM
QI-SCR	QTR_ADTL_INFO_B	YUQR9B
QM-SCR	QTR_COMPUTER_INFO	YUQR4
QN-SCR	QTR_CONTACT_INFO_C	YUQR3C
RPCSCR	RETRO_PAY_CONTROL	YURPC
RPMSCR	RETRO_PAY_METHOD_1	YURPMA
	RETRO_PAY_METHOD_2	YURPMB
RPPSCR	RETRO_PAY_PROCESS1	YURPPA
	RETRO_PAY_PROCESS2	YURPPB
STMSCR	SCHD_MEAL_PNLT	YT_SM
T1-SCR	TAX_SPECIFICATION	HIDX
TAWSCR	PNLT_WAIVERS	LTT
TE1SCR	TAX_WORK_LOCATION	UTE1
TE4SCR	TAX_WORK_LOCN	ML86
Multiple	EMPLOYMT_ACTIVITY	MLZC

Solution Series tables added/changed

The following table shows the Solution Series tables that have been added or amended since 5.1:

Tables Added/Changed since 5.1	Segment	Solution Series Form
DEPENDENT_GTL_CVG	L9X	59CSCR
EMP_MILITARY_DATE	ML6F	02-SCR
EMP_OVERTIME_PREM	L7D	OP-SCR
EMPLOYMT_ACTIVITY	MLZC	Multiple
FORMAL_EDUC_REQ	L6S	30HSCR
GRNS_ACH_HEAD	DCC8	GRKSCR
GRNS_C9_HED	DCC9	GRKSCR
GRNS_EE_ORD_PART1	L76	GR1SCR
GRNS_EE_ORD_PART2	L77	GR1SCR
GRNS_EE_ORD_PART3	L78	GR2SCR
GRNS_EE_ORD_PART4	L79	GR3SCR

Tables Added/Changed since 5.1	Segment	Solution Series Form
GRNS_EE_ORD_PART5	L7O	GR4SCR
GRNS_GARN_RULES	YUG07	GRGSCR
GRNS_NDI_REDUCT	YUG08	GRHSCR
GRNS_REQ_DED	YUG01	GRASCR
GRNS_REQ_DED_HED	YUG02	GRBSCR
GRNS_DED_PRI	YUG03	GRCSCR
GRNS_ST_ACH_HED	DCDA	GRLSCR
GRNS_SUPPORT_RULES	YUG06	GRFSCR
GRNS_SWTCH_OFF	YUG13	GRTSCR
GRNS_TAX_LEVY_RULE	YUG05	GRESCR
MISC_PAY_DETAILS	L8Z	PEMSCR
NAME_ADDR_REQ_NAME	L8Q	FFGSCR
PAY_PROCESS_OPT	DCAF	AF-SCR
PLCY_MEAL_PNLT	YT_PM	PTMSCR
PNLT_WAIVERS	LTT	TAWSCR
PR_TABLE_CTRL	YTZAW	AW-SCR
QTR_ADTL_INFO_B	YUQR9B	QI-SCR
QTR_COMPUTER_INFO	YUQR4	QM-SCR
QTR_CONTACT_INFO_C	YUQR3C	QN-SCR
REGISTER_OVERRIDES	DCCK	CK-SCR
RETROACTIVE_INFO	MLG6	HL-SCR ARPSUB
RETRO_PAY_CONTROL	YURPC	RPCSCR
RETRO_PAY_METHOD_1	YURPMA	RPMSCR
RETRO_PAY_METHOD_2	YURPMB	RPMSCR
RETRO_PAY_PROCESS1	YURPPA	RPPSCR
RETRO_PAY_PROCESS2	YURPPB	RPPSCR
SCHD_MEAL_PNLT	YT_SM	STMSCR
TAX_ARREARS_BLNCE	L6O	JTASCR
TAX_SPECIFICATION	HIDX	T1-SCR
TAX_WORK_LOCN	ML86	TE4SCR
TAX_WORK_LOCATION	UTE1	TE1SCR
V80_BENEFIT	LPD	PD-SCR

Tables obsoleted since 5.1:

Table	Segment
QTR_LOCAL_REPORTNG	YUQRL

A P P E N D I X F

Changes to Menu Records

In This Appendix

5.1 Menu Records

Previous Menu Record Layout

Menu records for The Solution Series have been changed for internationalization. The record now holds an alternate language version of the menu item title as well as the already existing primary language version. The record is a fixed length of 80 bytes.

Title	Size (bytes)	Position	Possible Values
Primary key	4	1–4 (inc)	"MMNP"
Level 0	1	5	"0" – "9"
Level 1	2	6–7 (inc)	"00" – "99"
Level 2	2	8–9 (inc)	"00" – "99"
Level 3	2	10–11 (inc)	"00" – "99"
Filler	12	12–23 (inc)	Blank text
Product Flag	1	24	" ", "W", "G", "N"
Program	6	25–31 (inc)	"EF-SCR", etc
Bitmap	1	32	"Y" or " "
Type	1	33	"C", "D", "H", "F", "S", "W"
Separator	1	34	"-" or blank
Clear Keys	1	35	"C"" or blank
Menu Title	40	36–75 (inc)	"Job Assignment In&quiry"
Help ID	4	76–79 (inc)	"8001" or blank

The 80th byte is used to terminate the record.

New Record Layout

The new record is 160 bytes long (2 x 80 bytes). This is accomplished by using two 'physical' records as one 'logical' record. Both records have the same key, and a suffix of either '1' for the first half of the record or a '2' for the second half. An example follows:

```
0....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
 MMNP0000420 1          G06-SCR F CJob Assignment In&quiry
 MMNP0000420 2           L'En&quête de Tâche de travail
```

Warning: The measurement characters above should not be used to calculate the position of the fields. To correctly calculate the position of a field, use the table that follows.

New Menu Record Layout Part 1

The table that follows shows the first 80 bytes of the record. New additions are highlighted.

Title	Size (bytes)	Position	Possible Values
Primary key	4	1–4 (inc)	"MMNP"
Level 0	1	5	"0" – "9"
Level 1	2	6–7 (inc)	"00" – "99"
Level 2	2	8–9 (inc)	"00" – "99"
Level 3	2	10–11 (inc)	"00" – "99"
Filler	1	12	Blank
New style record indicator	1	13	"1"
Filler	10	14–23 (inc)	Blank
Product Flag	1	24	" " for none, "W" for WebClient, "G" for GUI, or "N"" for none.
Program	6	25–30 (inc)	"EF-SCR", etc
Bitmap	1	31	"Y" or " "
Type	1	32	C, D, H, F, S, or W
Separator	1	33	"-" or blank
Clear Keys	1	34	"C" or blank
Menu Title	40	35–74 (inc)	"&Quick Hire"
Help ID	4	75–78 (inc)	"8001" or blank
Filler	1	79	blank

The 80th byte is used to terminate the record.

New Menu Record Layout Part 2

The concluding 'half' of the record (the second 80 bytes) is displayed below. New additions or differences are highlighted in bold.

Title	Size (bytes)	Position	Possible Values
Primary key	4	1–4 (inc)	"MMNP"
Level 0	1	5	"0" – "9"
Level 1	2	6–7 (inc)	"00" – "99"
Level 2	2	8–9 (inc)	"00" – "99"
Level 3	2	10–11 (inc)	"00" – "99"
Filler	1	12	Blank
New style record indicator	1	13	"2"
Filler	11	14–24 (inc)	Blank
Alternate Menu Title	40	25–64 (inc)	"L'En&quête de Tâche de travail"
Filler	15	65–79(inc)	Blank

The 80th byte is used to terminate the record.

5.2 Menu Records

The following tables detail the menu records added or amended in the 5.2 release.

New menu items

The following table gives details of menu records added for 5.2:

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Hot Key	Div line	Form	Sub menu	Help	Clear Keys	Web Dsply
HR	Employee Resourcing	Maintain Additional Employee Details	High Potential Employees	HP-SCR	H		*				
HR	Employee Development	Employee Skills, Competencies, Training	High Potential Employees	HP-SCR	H		*				
HR	HR Setup	Setup HR Rules	Organization to Rules Cross-Reference 2	BX-SCR	2		*			*	
			Automatic Employee Numbering	AUTSCR	M		*			*	
		Setup VETS-100 Rules (New submenu)			V			*			
		Setup VETS-100 Rules	Establish VETS-100 Headquarter Location	VTCSCR	Q		*				
			Establish VETS-100 Hiring Locations	VT-SCR	H		*				
			Reporting Information	HR9999	R	*			*		
HR	Position Administration	Position Details	Setup/Maintain Career Paths	MCPSCR	S		*				
			Key Positions	MKPSCR	Y		*				
		Workforce Planning (New submenu)			W	*					
		Workforce Planning	Setup/Maintain Career Paths	MCPSCR	S		*				
			High Potential Employees	HP-SCR	H		*				
			Key Positions	MKPSCR	Y		*				

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Hot Key	Div line	Form	Sub menu	Help	Clear Keys	Web Dsply
Benefits	Health/ Welfare Plan Enrollment/ Maint	Eligibility and Enrollment	Dependent Group Term Life Coverage	59CSCR	E		*				
Payroll	Employee Payroll	Maintain Employee Payroll Details	Overtime Premium	OP-SCR	R						
			Employee Dated HEDs	HHSSCR	E	*	*				
			Employee Dated Direct Deposits	H9SSCR	A		*				
			Employee Dated Hours and Pay Rate	H1SSCR	O		*				
		Maintain R.O.E. Information (New submenu)			R			*			
		Maintain Employee Garnishment Details (New submenu)			G			*			
		Maintain Employee Garnishment Details	Garnishment Admin 1 (Dates & Info)	GR1SCR	1		*				
			Garnishment Admin 2 (HED Info)	GR2SCR	2		*				
			Garnishment Admin 3 (Protections)	GR3SCR	3		*				
			Garnishment Admin 4 (Electronic Filing)	GR4SCR	4		*				
		View Employee Payroll Details (New submenu)			V			*			
		View Employee Payroll Details	Employee Dated HEDs	HSISCR	H		*				
		Time &Entry	Group Time Entry	DLG009	G						
Payroll	Payroll Setup Processing	Earnings/ Deductions/	Voluntary Deduction Rules	UX-SCR	V		*				

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Hot Key	Div line	Form	Sub menu	Help	Clear Keys	Web Dsply
		Accruals	Paycheck Estimator HEDs	UY-SCR	P		*				
		Payroll Processing Setup	Register Overrides	CK-SCR	R		*				
		Tax Information	Associated Tax Codes Table	URLSCR	O		*				N
		Garnishment Administration Rules (New submenu)			N			*			
		Garnishment Administration Rules	HEDs For Required Deduction HED Types	GRBSCR	H		*				
			HED Range for Garnishment Type	GRDSCR	E		*				
			Net Disposable Income Reduction HEDs	GRHSCR	N		*				
			Required Deduction HED Types	GRASCR	R	*	*				
			Deduction Priority	GRCSCR	D		*				
			Federal Tax Levy Protection Rules	GRESCR	F		*				
			Support Protection Rules	GRFSCR	S		*				
			Garnishment Protection Rules	GRGSCR	G		*				
			Switching off Garnishments	GRTSCR	W		*				
TA	Time and Attendance	Setup/Maintain Employee	Meal Penalty Waivers	TAWSCR	W		*				
		Set up TA Rules	Policy Meal Penalty Rules	PTMSCR	L		*			*	
			Schedule Meal Penalty Rules	STMSCR	M		*			*	

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Hot Key	Div line	Form	Sub menu	Help	Clear Keys	Web Dsply
TA	Time Entry (New submenu)				T						
TA	Time Entry	Employee Manager Time Entry (New submenu)			M				*		
		Employee Manager Time Entry	Time Entry Workflow Enabling	T934CR	W		*			*	
			Time Entry/HED Cross Reference	TATSCR	H		*			*	
			Time Entry Record Retention Parameters	ITRTCR	R		*			*	
			Department-Option List Mapping	GGTSCR	D		*			*	
			Payroll Administrator Inbox of Time Entries	ITMSCR	I		*			*	
			Payroll Administrator Approval/Process	GENTCR	A		*			*	
			Purge Time Entry Records	PURTCR	P		*			*	
			Local Administrator Setup	ITEADM	L		*			*	
Tools	User Tools	User Tools	Logon Duration Auditing	LOG000	D		*		*		
Tools	Administrator Tools	Administrator Tools	Purge FILE08	MNTF08	P		*				N
		Checklist Tools	Create/Modify a Checklist	WIZ001	C						N
			Remove a Checklist	DLG005	H						N
			Paused Checklist Details	MMPSCR	P	*	*				
			Edit Paused Checklists	MMPRSC	E		*				

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Hot Key	Div line	Form	Sub menu	Help	Clear Keys	Web Dsply
			View Completed Paused Checklist	MMPCSC	V		*				
			Delete Completed Paused Checklist	DELMMP	D		*				
Tools	Security Tools	Security Tools	Extended Security Setup	SECEXT	X		*				
			User Account Setup	SECUSR	T		*				
			Account Removal	SECRMV	V		*				

Amended menu items

The following table shows changes to the menu records or 5.2:

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Amendment
HR	Employee Resourcing	Maintain Additional Employee Details	Drivers License	25-SCR	Moved to after Non-Monetary Perquisites
			EEO-4 Reporting Information	EO4SCR	Changed places with Alternative Compensation Totals
	HR Setup	Setup EEO Rules	Establish EEO Hiring Locations	TX-SCR	Renamed from EEO Establishment Definition
			Establish EEO Parent Company Definition	TXCSCR	Renamed from EEO Parent Company Definition
	Position Administration	Position Details	Basic Details	M20SCR	Changed hot key designation from B to a
			Skills Required	M23SCR	Changed hot key designation from S to k
Benefits	Health/Welfare Plan Enrollment/Maint	Eligibility and Enrollment	Check EE Plan Eligibility	90-SCR	Changed hot key designation from E to k
Payroll	Employee Payroll	Maintain Employee Payroll Details	Reciprocal Tax Setup	JR-SCR	Deleted
		Maintain R.O.E. Information			New submenu
		Maintain R.O.E. Information	R.O.E. General Information	EI1SCR	Moved from Maintain Employee Payroll Details submenu
			R.O.E. Other Earnings	EI2SCR	Moved from Maintain Employee Payroll Details submenu
			R.O.E. Comments	EI3SCR	Moved from Maintain Employee Payroll Details submenu
		View Employee Payroll Details			New submenu
		View Employee Payroll Details	Earnings and Deductions	HHISCR	Removed 'View' from name and moved from Maintain Employee Payroll Details submenu

Menu (1)	Component (2)	Process (3)	Task (4)	Prog. name	Amendment
			Tax Information	JJISCR	Removed 'View' from name and moved from Maintain Employee Payroll Details submenu
			Labor Allocations	GGISCR	Removed 'View' from name and moved from Maintain Employee Payroll Details submenu
			Direct Deposit Information	H9ISCR	Removed 'View' from name and moved from Maintain Employee Payroll Details submenu
			Period End Date Records	PI-SCR	Removed 'View' from name and moved from Maintain Employee Payroll Details submenu
			Payment History/Labor Records	HL-SCR	Removed 'View' from name and moved from Maintain Employee Payroll Details submenu
	Payroll Setup Processing	Tax Information	Reciprocal Tax Mass Maintenance	JRMSCR	Deleted
Tools	Development Tools	System Control Repository Utilities	Rebuild T0P Cross Ref Records	ME5SCR	Changed 'O' to '0' in T0P

A P P E N D I X G

Large Number Changes to Fields

In This Appendix

Fields

The size of the following type 4 data fields has changed to accommodate the large numbers enhancement in 5.1.

42-HED-AMOUNT-CUR	PERM-01-V0	TAX-PREMIUM-MTD
42-HED-AMOUNT-MTD	PERM-01-V2	TAX-PREMIUM-QTD
42-HED-AMOUNT-QTD	PERM-02-V2	TAX-PREMIUM-YTD
42-HED-AMOUNT-YTD	PERM-03-V2	TAX-RESIDENT-MTD
ACCR-RESULT-BASE	PERM-04-V2	TAX-RESIDENT-QTD
COUNTER01	PERM-05-V2	TAX-RESIDENT-YTD
COUNTER02	PERM-06-V2	TAX-UNEMPLOY-MTD
COUNTER08	PERM-07-V2	TAX-UNEMPLOY-QTD
EMPLOYEE-WAGES-MTD	PERM-08-V2	TAX-UNEMPLOY-YTD
EMPLOYEE-WAGES-QTD	PERM-09-V2	TAX-WITHHELD-MTD
EMPLOYEE-WAGES-YTD	PERM-10-V2	TAX-WITHHELD-QTD
EMPLOYER-WAGES-MTD	PERM-11-V2	TAX-WITHHELD-YTD
EMPLOYER-WAGES-QTD	PERM-12-V2	TAX-WORK-MTD
EMPLOYER-WAGES-YTD	PERM-13-V2	TAX-WORK-QTD
ESS-PERM-01-V0	PERM-14-V2	TAX-WORK-YTD
EST-ANNUAL-INCOME	PERM-15-V2	TAXABLE-WAGES-MTD
HED-AMOUNT-CUR	PERM-16-V2	TAXABLE-WAGES-QTD
HED-AMOUNT-L-H	PERM-17-V2	TAXABLE-WAGES-YTD
HED-AMOUNT-L/H	PERM-21-V2	TEMP-01-V2
HED-AMOUNT-MTD	PERM-22-V2	TEMP-02-V2
HED-AMOUNT-QTD	PERM-23-V2	TEMP-03-V2
HED-AMOUNT-YTD	PERM-24-V2	TEMP-04-V2
HED-AMOUNT-CUR-12	PERM-25-V2	TEMP-05-V2
HED-AMOUNT-L/H-12	PERM-26-V2	TEMP-06-V2
HED-WORK-AMOUNT	SS-01-V2	TOTAL -PAY-MTD
LCF-PA-LCP-AMOUNT	SS-02-V2	TOTAL-PAY-QTD
		TOTAL-PAY-YTD

Hewitt

Installing and Configuring The Solution Series 5.2 (UNIX)

Document Issue: 1.2

Document Issue Status:	First Release
Document Issue Level:	1.2
Document Issue Date:	January 2007
Software Version:	5.2

Contents

Chapter 3   27

Chapter 4   51

Appendix F   155

P ART 1

Before You Begin...

In This Section

C H A P T E R 1

Introduction

In This Chapter

Welcome

This document provides detailed installation instructions for The Solution Series on a UNIX environment.

How to get additional help

If you cannot find the answers to your questions in this manual contact Customer Support who will be able to answer specific questions and give you general advice on training.

Please visit our web site www.hewitt.com/cyborg for the latest schedule of available courses and course descriptions.

Suggestions and feedback

We value your feedback on the performance support materials. Please forward any comments about this document to Customer Support.

Complete Product Installation Overview

Prerequisites

The software and hardware prerequisites for installing our products vary depending on your platform and the modules you purchased. Some third-party software must be purchased and installed before installing our products. To review hardware and software prerequisites for installing our products, follow these steps:

1. Access the Hewitt Cyborg home page
In the Address area at the top of your browser, type www.hewitt.com/cyborg and then press Enter.

2. Access the Customer Center
At the top of the home page click Customer Center Login.

3. Log in to the Customer Center
Click LOG IN, enter your User name and Password, and then either click OK or press Enter.

4. Select Product Updates
On the left pane of the page, click Product Updates.

5. Select prerequisites for the Product/Version
On the right side of the pane, click the product/version you want to view and their prerequisites.

P ART 2

Installing and Configuring The Solution Series Application Server

In This Section

C H A P T E R 2

Indexed Solution Series Installation and Configuration

In This Chapter

Indexed Server Installation and Configuration

This section provides detailed instructions for installing The Solution Series Application Server on a UNIX system. This is a technical section aimed at system administrators.

Note: *Throughout this section, we have suggested a naming convention for log files created when you execute the script. Entering a command like the following:*

```
rj jpayxtr
```

The command will run the script and create the log in the $log subdirectory. The log name will default to the name of the script, for example, jpayxtr.log. Please make sure you review all output after each script is run to identify and address any errors that may have occurred.

Deliverables

The following is included:

1	CD-ROM labeled The Solution Series 5.2 for UNIX
1	Installing and Configuring The Solution Series 5.2 (UNIX) (this guide)

☞   *Refer to Directory Contents for detailed information on scripts used and programs installed during the installation and the purposes they serve.*

Phase 1: Prepare for installation

Install and configure prerequisite software

It is assumed that you have already installed and configured the prerequisite software described in Chapter 1 on the server. These must be in place before beginning the installation.

Create user and file system

To create the user and file system for your installation, perform the following steps:

1. Check for, or create, a file system

Set a file system with at least 1GB of free space.

2. Create the installation user account

A user ID of 'cyborg' is required for the installation of the Cyborg Application Server (CAS) daemon. You must set up the permissions for this new account (for example 777 for the app directory and the files within it).

Note: *If you already have an existing cyborg account from a previous installation, re-use the existing cyborg user account.*

3. Log into the system

Log into the system using the given installation user account.

4. CBSVB and CBSVO Override

An override file is provided in the data directory called cbsv.ovr. The jpulcvn script which extracts the cobol programs use this file. If you have any user overrides for cbsvb or cbsvo, they must be added into this file prior to processing the script. In addition, two files are supplied:

- edi.ovr
- postcode.ovr.

If this functionality is required, they must be copied into cbsv.ovr.

Phase 2: Extract and Transfer Install Files

The Solution Series directory structure

The server software is delivered in two main directories:
`/cyborghome/app` and `/cyborghome/Cyborg52`. The name of the directory
`/cyborghome` and subdirectory `Cyborg52` is customizable during media installation.

Follow the appropriate steps listed below to copy the installation files and directory
structure to your server.

Note: *To allow each version of the software to operate independently using its own CAS, an
additional port other than 9888 can be set up for the Cyborg Application Server. It is
recommended that your `/cyborghome/app` directory exist only once no matter how many
environments you have installed for this version.*

```
                          cyborghome
        ┌──────────────┬──────────────┬──────────────┐
       app          cyborg52        cybtest         cybprod
        │              │               │               │
     server          data            data            data
        │              │               │               │
      trace           list            list            list
                       │               │               │
                      log             log             log
                       │               │               │
                      prog            prog            prog
                       │               │               │
                      runs            runs            runs
                       │               │               │
                      work            work            work
```

Extract install files to a PC

Insert the CD-ROM into The Solution Series Application Server machine. The 'Getting
Started' page starts automatically. Scroll through the page, then click the following link to
start the autoinstall:

Install The Solution Series application server

Follow the installation prompts. The following table tells you what information the
installation program will require. The center column, Options/Defaults contains defaults. If
you are not using defaults, complete the Your Entries/Selections column with information
needed for your organization's install.

Prompt	Options/Defaults	Your Entries/Selections
Destination	C:\Hewitt\Cyborg52\UNIX	

Prompt	Options/Defaults	Your Entries/Selections
Setup Type	*Complete *Custom	
UNIX	*AIX RS6000 *HP-UX *Sun Solaris	
Select Index or Relational	*Indexed *Relational (Oracle)	
UNIX Server HOSTNAME or IP Address	HOSTNAME	Enter IP Address or Hostname of UNIX Server
File System	/cyborghome/CYBORG52	

Transfer install files from the PC to the server

1. Verify system name and variables
FTP Command Script: ftpcmds_ST_idx.unx

The information that was entered during the copying of programs from the CD to the PC has been inserted into this file.

2. Run the file transfer program
Script used: jftp

Run the file transfer script from a command prompt on the Windows client. This will load the eCyborg files from the PC to the UNIX machine.

Note: *You will need the FTP capability enabled on both systems.*

Example of command line to run this script:

JFTP USERNAME

You will be prompted for the password. You can verify the FTP by checking the ftpupload.log file which will be generated in the same directory with the FTP script.

Note: *The following command is included in the FTP script:*

quote site chmod 755 jinstallst

If the following error: 'SITE command not implemented' is present in your ftpupload.log file, you must manually enable the access modifier to 755 on the jinstallst script before running it.

Extract server install files

Script used: jinstallst

This script should be run while logged in as CYBORG user. It extracts and installs the UNIX files onto the server.

Follow the installation prompts. The table following shows the information the install script will require along with the defaults used during the installation.

Prompt	Options/ Defaults
Enter the directory where the installation files were uploaded.	/installfiles/ Cyborg52_Install
Enter the top-level directory where the software will be installed.	/cyborghome
Please indicate whether you would like to install the base eCyborg product.	Enter Yes and press Enter.
Enter the name of the directory where The Solution Series will be installed.	CYBORG52
Please indicate whether you would like to install the Cyborg Application Server (CAS).	Enter Yes and press Enter.

The installation script will prompt you to verify that the information entered is correct and ask if you want to proceed with the installation. Once the files have been successfully installed, you may delete the CYBORG52_Install directory and all its files created by the FTP process.

Note: *The install program pipes error messages sent to the screen through user parameter $PAGER to control error messages from scrolling off the screen. Press a key when you see the -MORE- prompt.*

Set up user profile

The 'Cyborg' user needs its profile to include the updated values for the MicroFocus environment variables and $path. To set up the user profile, perform the following steps:

1. Modify Cyborg environment variable lines in the .profile
You need to modify the .profile identifying necessary variables for the environment. This is necessary for the compilation and execution of the background delivered source programs . Include the following lines in the .profile:

```
data=/cyborghome/CYBORG52/data      ; export data
work=/cyborghome/CYBORG52/work      ; export work
list=/cyborghome/CYBORG52/list      ; export list
prog=/cyborghome/CYBORG52/prog      ; export prog
runs=/cyborghome/CYBORG52/runs      ; export runs
log=/cyborghome/CYBORG52/log                ; export log
TERM=vt100                          ; export TERM
PATH=$PATH:/cyborghome/CYBORG52/runs ; export PATH
```

Note *The directory /cyborghome/CYBORG52 is the default directory for the product.*

2. Include Server Express environment variable lines in the .profile
You need to modify the .profile identifying necessary variables for the Micro Focus COBOL Compiler. This is necessary for the compilation and execution of the background delivered source programs. Please set the following environment variables:

- COBDIR - Specifies the directory that contains Server Express
- PATH - Specifies the directories to searched by all background programs, including the run-time system.
- LD_LIBRARY_PATH, LIBPATH or SHLIB_PATH, depending on your UNIX system. Specifies the directory fpr the UNIX system, cob and run-time system to search for shared libraries and callable shared objects.

Example:

For AIX RS6000

```
COBDIR=/usr/lpp/cobol                       ; export COBDIR
PATH=$COBDIR/bin:$PATH                       ; export PATH
PATH=/usr/ccs/bin                   ; export PATH
LIBPATH=$COBDIR/lib:$LIBPATH                 ; export LIBPATH
```

For Sun Solaris

```
COBDIR=/usr/lib/cobol                       ; export COBDIR
PATH=$COBDIR/bin:$PATH                       ; export PATH
LD_LIBRARY_PATH=$COBDIR/lib:$LD_LIBRARY_PATH ; export LD_LIBRARY_PATH
```

For HP-UX

```
COBDIR=/opt/lib/cobol                       ; export COBDIR
PATH=$COBDIR/bin:$PATH                       ; export PATH
SHLIB_PATH=$COBDIR/lib:$SHLIB_PATH           ; export SHLIB_PATH
```

Note   *Please refer to the Micro Focus installation guide for the settings for these variables.*

3.   Include location of C compiler in the PATH

 You need to add the file path of the C compiler to the .profile. For example:

For HP-UX

```
PATH=/opt/ansic/bin:$COBDIR/bin:/opt/bin/cobol:$PATH     ; export PATH
```

For AIX RS6000

```
PATH=/usr/vac/bin:$COBDIR/bin:/usr/lpp/cobolbin:$PATH     ; export PATH
```

For Sun Solaris

```
PATH=/usr/SUNWspro/bin:$COBDIR/bin:/usr/bin/cobol:$PATH  ; export PATH
```

4.   Refresh environment variables

Refresh the environment so the logical names take effect by executing the following:

. ./.profile (dot-space-dot-slash-dot-profile)

Verify compiler versions

The delivered binary programs were compiled on the certified platform versions and are not guaranteed to run on earlier versions. To verify that ServerExpress environment variables are set correctly, perform the following steps:

1. **Execute the cob command**

   To verify that ServerExpress environment variables are set correctly, execute the following command:

   ```
   cob
   ```

   The response should be:

   ```
   I see no work
   ```

   Note: *If you do not receive this response, verify that your ServerExpress environment variables are set correctly.*

2. **Verify the version of the installed compilers**

   To verify what version of ServerExpress is installed, execute the following command:

   ```
   cat $COBDIR/etc/cobver
   ```

   This will show you the version and revision level of the COBOL compiler installed. To obtain the C version, perform the following command for your operating system:

   For Sun Solaris

   ```
   $ cc -V 2>&1 | head -1
   ```

   For AIX RS6000

   ```
   $ lslpp -L|grep ibmc**.cmp|cut -c30-34|head-
   ```

   For HP-UX

   ```
   $ what /usr/bin/cc|head -2|tail -1
   ```

   Note: *These examples are case-sensitive.*

Check special requirements

1. **Check output**

   Each time a script is run, check all output before another one is executed. For example, verify the output of the compiles and/or audit reports.

2. **Server Express**

   Server Express can be installed as a 32 or 64 bit product. This release has been certified using the 32 bit compiler. To ensure your implementation is using the 32 bit compiler, include the following in all the .profiles and the CAS script:

   COBMODE=32 ; export COBMODE

3. **Compile all COBOL programs with the align(8) compiler directive**

   Insert the align (8) compiler directive into your existing compile jobs. The following is a sample of how you include this compiler directive:

   ```
   cob –xv –C align(8) nobound ibmcomp noosvs notrunc –N nocheck noboundopt
   $prog/{program}.cob -o $prog/{program}
   ```

Phase 3: Compile Batch Programs and Build FILE01

Perform the following steps to install the batch and online programs and create the random System Control Repository.

Note: *This portion of the installation must be performed locally on the server.*

Jobs are run with the following syntax:

rj jobname

The rj prefix runs the job and creates a log. It is important to note that rj runs the job in the background. When a job is submitted, the first part of the log is displayed on the console. Control is then returned to the command prompt while the job continues to run in the background. Use the 'tail' command to examine the end of the log file, until the Job Complete message displays. Review the output after each script has completed to identify any errors that may have occurred. Review the output after each script has completed to identify any errors that may have occurred.

Extract and compile all cybmst programs

Script used: jxcybmst
To extract and compile all cybmst COBOL programs (P9CNVT, P2EDIT, P4CALC, P5PRNT, P7COMP, 04CALC) and cbsvrft, execute the jxcybmst script from the $runs subdirectory. For example:

rj jxcybmst

Review the log, then the cybmst.03 list file in the $list subdirectory to determine if there were any errors.

Compile and link sort programs

Script used: jcmpsort
To compile p10sort.cob, p45sort.cob, p80sort.cob, p80copy.cob, and pfssort.cob execute the jcmpsort script from the $runs subdirectory. For example:

rj jcmpsort

Review the log to determine if there were any errors.

Compile and link the delivered cbsvb

Script used: jcmpcvbn
To compile the non-relational batch program cbsvb as delivered, execute the jcmpcvbn script from the $runs subdirectory. For example:

rj jcmpcvbn

Review the log to determine if there were any errors.

Compile and link the US Quarterly Processor

Script Used: jxp5qtr

To compile the script p5qtr as delivered, execute jxp5qtr from the $runs subdirectory. For example:

rj jxp5qtr

Review the log to determine if there were any errors.

☞ *Refer to Using the Quarterly Processor for instructions on selecting generators, configuring the processor, and generating quarterly report output.*

Create System Control Repository

Script used: jdemo01

This procedure creates the System Control Repository. Execute the jdemo01 script. For example:

rj jdemo01

Review the log, then the demo.03 list file in the $list subdirectory to determine if there were any errors. If you receive any 'RELOAD NOT FOUND' messages, disregard these messages. For example:

```
HEWITT ASSOCS  - RELOAD PROGRAM CHECK
-----------------------------------
P CYBADT ******** RELOAD NOT FOUND ********
P CYBHL  ******** RELOAD NOT FOUND ********
P CYBP15 ******** RELOAD NOT FOUND ********
P CYBPZQ ******** RELOAD NOT FOUND ********
P CYBRCI ******** RELOAD NOT FOUND ********
P CYBSCK ******** RELOAD NOT FOUND ********
P CYBSEC ******** RELOAD NOT FOUND ********
P CYBWCI ******** RELOAD NOT FOUND ********
P CYBWRK ******** RELOAD NOT FOUND ********
P CYBWZQ ******** RELOAD NOT FOUND ********
P CYBX02 ******** RELOAD NOT FOUND ********
P EXCTRL ******** RELOAD NOT FOUND ********
P QMCTRL ******** RELOAD NOT FOUND ********
P RDEMRL ******** RELOAD NOT FOUND ********
P RDEMRQ ******** RELOAD NOT FOUND ********
P RDTBPL ******** RELOAD NOT FOUND ********
P RTCTRL ******** RELOAD NOT FOUND ********
P TBLENT ******** RELOAD NOT FOUND ********
P TBLINQ ******** RELOAD NOT FOUND ********
P TBLUPD ******** RELOAD NOT FOUND ********
P TBLVER ******** RELOAD NOT FOUND ********
P V-NAME ******** RELOAD NOT FOUND ********
P WPTM   ******** RELOAD NOT FOUND ********
```

Phase 4: Compile CBSV and CYBIO

Extract cbsv programs

Script used: jpulcvn

For example:

rj jpulcvn

Review the log and then the pulcvn.lis list file in the $list subdirectory to determine if there were any errors.

Compile and link all CBSV programs

Script used: jcmpcvn

For example:

rj jcmpcvn

Review the log to determine if there were any errors.

Compile and link CYBIO programs

Script used: jcmpcbio

To compile and link the CYBIO programs, execute the jcmpcbio script from the $runs subdirectory. For example:

rj jcmpcbio

Review the log to determine if there were any errors.

Phase 5: Create Test P20IN Batch Master

Extract Report Generators

Script used: jp20strt (U.S.)
                       jp20strc (Canada)

To extract the report generators from CYBMST and create the P20IN Batch Master File, execute the jp20strt script from the $runs subdirectory with p9strt and p9cbsv as input.

Input files

p05t80   p9strt

p05t81   p9cbsv (U.S.)
                                       p9cbsvc (Canada)

The P2EDIT, P4CALC, and P5PRNT programs are processed. For example:

rj jp20strt

Review the log and then all the output listings in the $list subdirectory to determine if there were any errors.

Output listings

audit2.trl
p20strt.03
transload

Create Test Employee Database

Script used: jpaymrg

To create a test Employee Database, execute the jpaymrg script from the $runs subdirectory. For example:

rj jpaymrg

Review the log and then the paymrg.03 list file in the $list subdirectory to determine if there were any errors.

Phase 6: Create Test Employee Database

Populate database with test data

To populate the database with test data, perform the following steps:

1. **Load test companies**
   Script used: jprdemo (U.S.)
   jprdemoc (Canada)

   This procedure loads the payroll test data into the Employee Database (Master File: FILE02). Execute the jprdemo script from the $runs subdirectory. For example:

   rj jprdemo

   Review the log and then the prdemo.03 list file in the $list subdirectory to determine if there were any errors.

2. **Load HR test data**
   Script used: jhrdemo (U.S.)
   jhrdemoc (Canada)

   This procedure loads HR test data into the Employee Database (Master File: FILE02). Execute the jhrdemo script from the $runs subdirectory. For example:

   rj jhrdemo

   Review the log and then the hrdemo.03 list file in the $list subdirectory to determine if there were any errors.

3. **Build alternate keys**
   Script used: jbldaky

   To build or rebuild the Employee Name Alternate Key, run the jbldaky script located in the $runs directory.

Note: *This script may be altered to accommodate your build of the alternate keys once your data has been added to the system.*

   For example:

   rj jbldaky

   Review the log and then the bldaky.03 list file in the $list subdirectory to determine if there were any errors.

Extract full payroll data

   Script used: jpayxtr

   To update the P20IN Batch Master File with the demo test data, execute the jpayxtr script from the $runs subdirectory.

   The organization and employee data will be extracted from the online Employee Database and placed in FILE12. This will be the new P20IN Batch Master File.

FILE10 (payxtr10), which contains time entries and adjustments, is also created. payxtr10 becomes input to the jpayrun as p05t81. For example:

rj jpayxtr

Review the log and then the payxtr.03 list file in the $list subdirectory to determine if there were any errors.

Complete a payroll run

Script used: jpayrun

Complete a payroll run to apply taxes (TAXFILE) and the time entries and adjustments (payxtr10) to the p20in file. Execute the jpayrun script from the $runs subdirectory with p05t80 and p05t81 as input. The P2EDIT, P4CALC, and P5PRNT programs will be processed.

Note: *Verify that the TAXFILE being applied is the current tax related regulatory bulletin from CUBBS (Cyborg Users Bulletin Board).*

☞ *Refer to Accessing CUBBS in the knowledgebase or Technical Administration guide for detailed instructions for logging onto CUBBS.*

Input files

p05t80	TAXFILE (US)
	TAXFILEC (Canada)
p05t81	payxtr10

Note: *If you are a Canadian customer, modify the script to extract TAXFILEC instead of the TAXFILE.*

For example:

rj jpayrun

Review the log, then all the output listings in the $list subdirectory to determine if there were any errors.

Output listings

auditrl1.lis
checknum.lis
payslips.lis
depslips.lis
combreg.lis
transload.lis

Perform a payroll maintenance run

Script used: jmntrun

Perform a payroll maintenance run to create pay history and labor records and apply check numbers to the newly created history records on the p20in Batch Master File. Execute the

jmntrun script from the $runs subdirectory. The P2EDIT, P4CALC, and P5PRNT programs will be processed. For example:

rj jmntrun

Review the log, then the transload2.lis and auditrl2.lis list files in the $list subdirectory to determine if there were any errors.

Update the Employee Database

Script used: jpaymrg

To create a new random Employee Database, execute the jpaymrg script from the $runs subdirectory. For example:

rj jpaymrg

Review the log, then the paymrg.03 list file in the $list subdirectory to determine if there were any errors.

Phase 7: Extract HR reports

Extract HR Reports

Script Used: jreport

To extract Human Resource batch reports, execute the jreport script from the $runs subdirectory. For example:

rj jreport

Review the log, then the rtprnt.03 list file in the $list subdirectory to determine if there were any errors.

Phase 8: Apply System Control Repository Menu Additions

If you plan on implementing the Report Launcher, Process Monitor, and Desktop Document Interface facilities on the client, the you must apply additional menu items to the System Control Repository (FILE01).

Build report menus

Script used: jrptmnu

To build a menu containing the delivered report groups, run jrptmnu located in the $runs directory. For example:

rj jrptmnu

Review the log, then the rptmnu.03 list file in the $list subdirectory to determine if there were any errors.

Apply menu and checklist updates for Document Launcher and Enhanced Payroll Reporting

Script used: jmainti

To apply menu items for online user access to batch processes, run the jmainti script from the $runs directory, using the $data/eprddi05 file as FILE05 input. For example:

rj jmainti

Review the log, then the mainti.03 list file in the $data subdirectory to determine if there were any errors.

C H A P T E R   3

Relational Solution Series Installation and Configuration

In This Chapter

Introduction

This section provides detailed instructions for installing The Administrative Server on a UNIX system. This is a technical section aimed at system administrators.

Note: *Throughout this section, we have suggested a naming convention for log files created when you execute the script. Entering a command like the following:*

```
rj jpayxtr
```

The command will run the script and create the log in the $log subdirectory. The log name will default to the name of the script, for example, jpayxtr.log. Please make sure you review all output after each script is run to identify and address any errors that may have occurred.

Deliverables

The following is included:

1	CD-ROM labeled The Solution Series 5.2 for UNIX
1	Installing and Configuring The Solution Series 5.2 (UNIX) (this guide)

☞   *Refer to Directory Contents for detailed information on scripts used and programs installed during the installation and the purposes they serve.*

Phase 1: Prepare for installation

Install and configure prerequisite software

It is assumed that you have already installed and configured the prerequisite software described in Chapter 1 on the server. These must be in place before beginning the installation.

Create user and file system

To create the user and file system for your installation, perform the following steps:

1. Check for, or create, a file system

Set a file system for CYBORG with at least 1GB of free space.

2. Create the installation user account

A user ID of Cyborg is required for the installation of the Cyborg Application Server (CAS) daemon.You must set up the permissions for this new account (for example 777 for the app directory and the files within it).

Note: *If you already have an existing cyborg account from a previous installation, re-use the existing cyborg user account.*

3. Log into the system

Log into the system using the Cyborg user account.

Note: *The Cyborg user account information will be needed for later steps.*
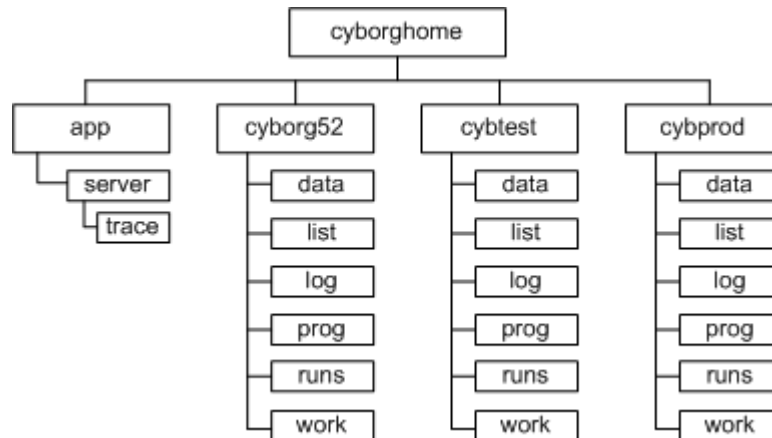
Phase 2: Extract and Transfer Install Files

The Solution Series directory structure

The server software is delivered in two main directories: `/cyborghome/app` and `/cyborghome/Cyborg52`. The name of the directory `/cyborghome` and subdirectory `Cyborg52` is customizable during media installation.

Follow the appropriate steps listed below to copy the installation files and directory structure to your server.

Note: *To allow each version of the software to operate independently using its own CAS, an additional port other than 9888 can be set up for the Cyborg Application Server. It is recommended that your `/cyborghome/app` directory exist only once no matter how many environments you have installed for this version.*

Extract install files to a PC

Insert the CD-ROM into The Solution Series Application Server machine. The Getting Started page starts automatically. Scroll through the page, then click the following link to start the autoinstall:

Install The Solution Series application server

Follow the installation prompts. The table following tells you what information the installation program will require. The center column, Options/Defaults contains the defaults. If you are not using the defaults, complete the Your Entries/Selections column with information needed for your organization's install.

Prompt	Options/Defaults	Your Entries/Selections
Destination	C:\Hewitt\Cyborg52\UNIX	

Prompt	Options/Defaults	Your Entries/Selections
Setup Type	*Complete *Custom	
UNIX	*AIX RS6000 *HP-UX *Sun Solaris	
Select Index or Relational	*Indexed *Relational (Oracle)	
UNIX Server HOSTNAME or IP Address	HOSTNAME	Enter IP Address or Hostname of UNIX Server:
File System	/*cyborghome*/CYBORG52	

The installation program will prompt you when it is complete.

Transfer install files from the PC to the server

1.   Verify system name and variables
FTP Command Script: ftpcmds_st_ora.unx
The information that was entered during the copying of programs from the CD to the PC has been inserted into this file.

Note: The script is located in the destination defined in the extract step.

2.   Run the file transfer program
Script used: jftp
Run the file transfer script from a command prompt on the Windows client. This script will load the Solution Series files from the PC to the UNIX machine.

Note: The script is located in the destination defined in the extract step and you will need to have the FTP capability enabled on both systems.

Example of command line to run this script:

JFTP USERNAME

You will be prompted for the password. You can verify the ftp by checking the ftpupload.log file which will be generated in the same directory with the ftp script.

Note: The following command is included in the ftp script:

quote site chmod 755 jinstallst

If the following error SITE command not implemented is present in you ftpupload.log file, you must manually enable the access modifier to 755 on the jinstallst script before running it.

Extract server install files

Script used: jinstallst

This script should be run while logged in as CYBORG user. It extracts and installs the UNIX files onto the server. Follow the installation prompts. The table following shows the information the install script will require along with the defaults used during the installation.

Prompt	Options/ Defaults
Enter the directory where the installation files were uploaded.	/installfiles/ Cyborg52_Install
Enter the top-level directory where the software will be installed.	/cyborghome
Please indicate whether you would like to install the base eCyborg product.	Enter Yes and press Enter.
Enter the name of the directory where The Solution Series will be installed.	CYBORG52
Please indicate whether you would like to install the Cyborg Application Server (CAS).	Enter Yes and press Enter.

The installation script will prompt you to verify that the information entered is correct and ask if you wish to proceed with the installation. Once the files have been successfully installed, you may delete the CYBORG52_Install directory and all its files created by the FTP process.

Note: The install program pipes error messages sent to the screen through user parameter $PAGER to control error messages from scrolling off the screen. Press a key when you see the -MORE- prompt.

Set up user profile

Every batch user must include the updated values for the MicroFocus environment variables and $path (to include the Cyborg user). To set up the user profile, perform the following steps:

1.   Modify environment variable lines in the .profile

You need to modify the .profile identifying necessary variables for the environment. This is necessary for the compilation and execution of the background delivered source programs v5.2. Include the following lines in the .profile.

```
data=/cyborghome/CYBORG52/data     ; export data
work=/cyborghome/CYBORG52/work     ; export work
list=/cyborghome/CYBORG52/list     ; export list
prog=/cyborghome/CYBORG52/prog     ; export prog
runs=/cyborghome/CYBORG52/runs     ; export runs
log=/cyborghome/CYBORG52/log                 ; export log
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252        ; export NLS_LANG
TERM=vt100                                    ; export TERM
PATH=$PATH:/cyborghome/CYBORG52/runs          ; export PATH
```

Note: *The directory /cyborghome/CYBORG52 is the default directory.*

2. Include Server Express environment variable lines in the .profile

You need to modify the .profile identifying necessary variables for the Micro Focus COBOL Compiler. This is necessary for the compilation and execution of the background delivered source programs v5.2. Please set the following environment variables:

- COBDIR - Specifies the directory that contains Server Express
- PATH - Specifies the directories to searched by all background programs, including the run-time system.
- LD_LIBRARY_PATH, LIBPATH or SHLIB_PATH, depending on your UNIX system. Specifies the directory for the UNIX system, cob, and run-time system to search for shared libraries and callable shared objects.

Examples:

For AIX RS6000

```
COBDIR=/usr/lpp/cobol                       ; export COBDIR
PATH=$COBDIR/bin:$PATH                       ; export PATH
LIBPATH=$COBDIR/lib:$LIBPATH                 ; export LIBPATH
```

For Sun Solaris

```
COBDIR=/usr/lib/cobol                        ; export COBDIR
PATH=$COBDIR/bin:$PATH                        ; export PATH
LD_LIBRARY_PATH=$COBDIR/lib:$LD_LIBRARY_PATH ; export LD_LIBRARY_PATH
```

For HP-UX

```
COBDIR=/opt/lib/cobol                        ; export COBDIR
PATH=$COBDIR/bin:$PATH                        ; export PATH
SHLIB_PATH=$COBDIR/lib:$SHLIB_PATH           ; export SHLIB_PATH
```

3. Include location of C compiler in the PATH

You need to add the file path of the C compiler to the .profile. For example:

For AIX RS6000

```
PATH=/usr/vac/bin:$COBDIR/bin:/usr/lpp/cobolbin:$PATH    ; export PATH
```

For Sun Solaris

```
PATH=/usr/SUNWspro/bin:$COBDIR/bin:/usr/bin/cobol:$PATH  ; export PATH
```

For HP-UX

```
PATH=/opt/ansic/bin:$COBDIR/bin:/opt/bin/cobol:$PATH     ; export PATH
```

4. Include Oracle environment variable lines in the .profile

You need to modify the .profile identifying necessary variables for Oracle. This is necessary for the compilation and execution of the background delivered source programs v5.2. Please set the following environment variables:

```
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252            ;export NLS_LANG
ORACLE_ADMIN=                      ;export ORACLE_ADMIN
ORACLE_BASE=                       ;export ORACLE_BASE
ORACLE_HOME=                       ;export ORACLE_HOME
```

```
ORACLE_PATH=                                    ;export ORACLE_PATH
ORACLE_SID=                                     ;export ORACLE_SID
```

5.   Refresh environment variables

Refresh the environment so the logical names take effect by executing the following:

. ./.profile (dot-space-dot-slash-dot-profile)

Verify compiler versions

The delivered binary programs were compiled on the certified platform versions and are not guaranteed to run on earlier versions. To verify that ServerExpress environment variables are set correctly, perform the following steps:

1.   Execute the cob command

To verify that ServerExpress environment variables are set correctly, execute the following command:

```
cob
```

The response should be:

```
I see no work
```

Note: If you do not receive this response, verify that your ServerExpress environment variables are set correctly.

2.   Verify the version of the installed compilers

To verify what version of ServerExpress is installed, execute the following command:

```
cat $COBDIR/etc/cobver
```

This will show you the version and revision level of the COBOL compiler installed. To obtain the C version, perform the following command for your operating system:

For Sun Solaris

```
$ cc -V 2>&1 | head -1
```

For AIX RX6000

```
$ lslpp -L|grep ibmc**.cmp|cut -c30-34|head-
```

For HP-UX

```
$ what /usr/bin/cc|head -2|tail -1
```

Note: These examples are case-sensitive.

Check special requirements

1.   Check output

Each time a script is run, check all output before another one is executed. For example, verify the output of the compiles and/or audit reports.

2.   **Server Express**

Server Express can be installed as a 32 or 64 bit product. This release has been certified using their 32 bit compiler. To assure your implementation is using the 32 bit compiler, include the following in the all .profiles and the cas script:

COBMODE = 32 ; export COBMODE

3.   **ORACLE Character Set**

The we8iso8859p1 character set does not support all characters. Using this character set, you will experience operational errors during installation. When you create the database, we recommend choosing the following:

- WE8MSWIN1252 as the database character set
- AL16UTF16 as the national character set

In order to determine this, enter the following command. This command lists the language-relevant parameters of your database. Among these are the contents of the variables verify the contents of NLS_CHARACTERSET and NLS_NCHAR_CHARACTERSET, which were selected when the database was created.

SELECT * FROM sys.props$

WHERE name LIKE 'NLS%';

4.   **Compile all COBOL programs with the 'align(8)' compiler directive**

Insert the align (8) compiler directive into your existing compile jobs. The following is a sample of how you include this compiler directive:

```
cob –xv –C "align(8) nobound ibmcomp noosvs notrunc" –N "nocheck noboundopt"
$prog/{program}.cob –o $prog/{program}
```

Phase 3: Build The Solution Series Environment

Perform the following steps to install the batch and online programs and create the random System Control Repository.

Note: *This portion of the installation must be performed locally on the server. By preceding the batch job with 'rj' you run the script and create the log. Be sure to review output after you run each script to identify and address any errors that may have occurred.*

Extract and compile all cybmst programs

Script used: **jxcybmst**

To extract and compile all cybmst COBOL programs (P9CNVT, P2EDIT, P4CALC, P5PRNT, P7COMP, O4CALC) and cbsvrft, execute the jxcybmst script from the `$runs` subdirectory. For example:

rj jxcybmst

Review the log, then the cybmst.03 list file in the `$list` subdirectory to determine if there were any errors.

Compile and link sort programs

Script used: **jcmpsort**

To compile p10sort.cob, p45sort.cob, p80sort.cob, p80copy.cob, and pfssort.cob, execute the jcmpsort script from the `$runs` subdirectory. For example:

rj jcmpsort

Review the log to determine if there were any errors.

Compile and link the US Quarterly Processor

Script used: **jxp5qtr**

To compile the batch program p5qtr as delivered, execute the jxp5qtr script from `$runs`. For example:

rj jxp5qtr

Review the log to determine if there were any errors.

☞ *Refer to Using the Quarterly Processor documentation for instructions on selecting generators, configuring the processor, and generating quarterly report output.*

Compile rdbpgm0.cob

Script Used: **jcmprdb0**

To compile the rdbpgm0.cob program, execute the jcmprdb0 script from the `$runs` subdirectory. For example:

rj jcmprdb0

Review the log to determine if there were any errors.

Execute the CASE tool

Script Used: jcrtpgms

1. Modify the control record in jcrtpgms

Modify the control record in jcrtpgms to include the datafile path, database connect string, and the tablespace indicator to uniquely identify this environment.

Important!: The database must be created by the Database Administrator. In addition to the rdbpgm1 program created in this step, all cbsv programs will contain the connect clause for the database.

2. Execute jcrtpgms

Execute the jcrtpgms script from the $runs subdirectory. For example:

rj jcrtpgms

Review the log to determine if there were any errors.

Execute the make command

To be able to precompile, compile, and link The Solution Series relational programs, execute the make commands and test the sample1 program delivered by ORACLE.

☞ *Please refer to the ORACLE installation guide for information on the name and location of the make file.*

Log in as the ORACLE administrator and execute the make commands as in the following example (actual commands may differ according to compiler):

```
cd $ORACLE_HOME/precomp/demo/procob2
make -f demo_procob_32.mk -n sample1 > $runs/cyborg.mk
```

Note: *If the program above generated an 'Undefined symbol: .pthread_yield' error, this is due to the fact that pthread_yield() was removed from the Posix (XOpen Version 5) standard. IBM provides a compatibility library for vendors who still use pthread_yield(). You will still need to create the cyborg.mk file; however, you will need to modify the cyborg.mk file to include a reference to '-lpthreads_compat' near the start of the file.*

ORACLE linkage

The ORACLE linkage step (cyborg.mk) created by the make command will be needed for the following relational scripts:

jcmpsubr
jcmprdb1
jxo4calr
jcmpcvr
jcmpcvbr

Pre-compile, compile, and link rdbpgm1

1. Add Oracle linkage before running this script

This delivered script includes a sample ORACLE linkage step (underlined in the script). This linkage step must be replaced by the linkage step created by executing the ORACLE make command in Execute the make command.

```
echo \n\n\t  JCMPRDB1 IN PROGRESS  \n
echo \n\n\t  CHECKING FOR FILES \n
test -f $prog/rdbpgm1.pco || echo \n\t rdbpgm1.pco does not exist
cd $prog
procob32 ireclen=132 oreclen=132 select_error=no litdelim=apost mode=ansi
iname=rdbpgm1.pco oname=rdbpgm1.cob
cob -xv -C IBMCOMP NOOSVS NOBOUND NOTRUNC ALIGN(8) -N nocheck noboundopt -o
rdbpgm1 rdbpgm1.cob \
\
-L /oracle/OraHome1/lib32/ /oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh
cat /oracle/OraHome1/lib32/ldflags    cat
 /oracle/OraHome1/lib32/sysliblist  -lm  -lpthread -lpthread
/
cd $runs
echo \n\n\t JCMPRDB1 COMPLETE
```

For HP-UX and Sun Solaris, procob32 is executed. For AIX RS6000, procob or procob32 is executed.

2. Execute jcmprdb1

To pre-compile, compile, and link the program rdbpgm1, execute the jcmprdb1 script from the $runs subdirectory. For example:

rj jcmprdb1

Review the log to determine if there were any errors.

Create the tablespaces, tables, indexes, and views

Script Used: jcrtcyb

To execute the SQL statements defined in rdbpgm1 and create the tablespaces, tables, indexes, and views, execute the jcrtcyb script from the $runs subdirectory. There is no output from this run, but you or your database administrator can execute the SQL statements to verify the tablespaces, tables, indexes, and views have been created. For example:

rj jcrtcyb

Review the log to determine if there were any errors.

Pre-compile, compile, and link rdbpgma through rdbpgmh

1. Add Oracle linkage before running this script

This delivered script includes a sample ORACLE linkage step (underlined in the script). This linkage step must be replaced by the linkage step created by executing the ORACLE make command in Execute the make command. For example:

```
echo \n\n\t  JCMPSUBR IN PROGRESS \n
```

```
echo \n\n\t  CHECKING FOR FILES \n
test -f $prog/rdbpgma.pco || echo \n\t rdbpgma.pco does not exist
test -f $prog/rdbpgmb.pco || echo \n\t rdbpgmb.pco does not exist
test -f $prog/rdbpgmc.pco || echo \n\t rdbpgmc.pco does not exist
test -f $prog/rdbpgmd.pco || echo \nrdbpgmd.pco does not exist
test -f $prog/rdbpgme.pco || echo \n\t rdbpgme.pco does not exist
test -f $prog/rdbpgmf.pco || echo \n\t rdbpgmf.pco does not exist
test -f $prog/rdbpgmg.pco || echo \n\t rdbpgmg.pco does not exist
test -f $prog/rdbpgmh.cob || echo \n\t rdbpgmh.cob does not exist
cd $prog

for file in rdbpgma rdbpgmb rdbpgmc rdbpgmd rdbpgme rdbpgmf rdbpgmg
do
echo \n\n\t COMPILATION OF ${file} IN PROGRESS \n
procob32  iname=${file}.pco oname=${file}.cob ireclen=132 oreclen=132
select_error=no litdelim=apost mode=ansi
cob -xv -C ibmcomp noosvs nobound notrunc align(8) -N nocheck noboundopt -o
${file} ${file}.cob \
\
-L/oracle/OraHome1/lib32/ /oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh cat
/oracle/OraHome1/lib32/ldflags    cat
/oracle/OraHome1/lib32/sysliblist`  -lm  -lpthread -lpthread
/
echo \n\n\t COMPILATION OF ${file} COMPLETE \n
done

echo \n\n\t  COMPILATION OF RDBPGMH IN PROGRESS \n
cob -xv -C ibmcomp noosvs nobound notrunc align(8) -N nocheck noboundopt
$prog/rdbpgmh.cob -o $prog/rdbpgmh
echo \n\n\t  COMPILATION OF RDBPGMH COMPLETE \n
cd $runs
echo \n\n\t  JCMPSUBR COMPLETE
```

For HP-UX and Sun Solaris, procob32 is executed. For AIX RS6000, procob or procob32 is executed.

2.   Execute jcmpsubr

To pre-compile, compile, and link the programs rdbpgma through rdbpgmh, execute the jcmpsubr script from the $runs subdirectory. For example:

rj jcmpsubr

The following table lists and describes each program generated by the CASE tool.

Program	Description
rdbpgma	The subroutine that handles inserting a new row in a table.
rdbpgmb	The subroutine that handles selecting data from a row in a table and passing it to cbsv.
rdbpgmc	The subroutine that handles updating values in an existing row.
rdbpgmd	The subroutine that handles deleting an existing row from a table.
rdbpgme	The subroutine called when a PAYMRG 171 process is run. It removes all rows from all tables in preparation for reinsertion of data from the P20 file. It also disables, then re-enables, all indexes (where applicable).

Program	Description
rdbpgmf	The subroutine called when a PAYMRG 222 process is run. It deletes all rows from the appropriate tables that belong to the Organizations being paid.
rdbpgmg	The subroutine that cursors through the database and rebuilds the IDX records on the database.
rdbpgmh	The subroutine that provides segment and segment key length for each segment and location of data and data type within each segment.

Review the log to determine if there were any errors.

Extract, compile, and link o4calc

1.   Add Oracle linkage before running this script

This delivered script includes a sample ORACLE linkage step (underlined in the job). This linkage step must be replaced by the linkage step created by executing the ORACLE make command in Execute the make command.

```
echo \n\n\t  JXO4CALR IN PROGRESS
echo \n\n\t  CHECKING FOR FILES \n
test -f $prog/p9cnvt       || echo \n\t p9cnvt does not exist
test -f $data/cybmst       || echo \n\t cybmst does not exist
test -f $data/vers80.ovr   || echo \n\t vers80.ovr does not exist
test -f $prog/rdbpgma.o    || echo \n\t rdbpgma.o does not exist
test -f $prog/rdbpgmb.o    || echo \n\t rdbpgmb.o does not exist
test -f $prog/rdbpgmc.o    || echo \n\t rdbpgmc.o does not exist
test -f $prog/rdbpgmd.o    || echo \n\t rdbpgmd.o does not exist
test -f $prog/rdbpgmh.o    || echo \n\t rdbpgmh.o does not exist

echo \n\n\t  CREATION OF O4CALC (RELATIONAL) IN PROGRESS \n
echo O4CALC      |ISEV@PCYd&  24       MICRO-FOCUS.> $work/o4calc.04
FILE1=$prog/o4calc.pco  ;  export FILE1
PRINT1=$list/o4calc.03  ;  export PRINT1
P05RDR=$work/o4calc.04  ;  export P05RDR
CYBMST=$data/cybmst     ;  export CYBMST
$prog/p9cnvt

cd $prog
echo \n\n\t COMPILATION OF O4CALC IN PROGRESS \n
procob32  iname=o4calc.pco oname=o4calc.cob ireclen=132 oreclen=132
select_error=no litdelim=apost mode=ansi
cob -xv -C IBMCOMP NOOSVS NOBOUND NOTRUNC ALIGN(8) -N NOCHECK NOBOUNDOPT -o o4calc
o4calc.cob rdbpgma.o rdbpgmb.o rdbpgmc.o rdbpgmd.o rdbpgmh.o \
\
-L/oracle/OraHome1/lib32/ /oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh cat
/oracle/OraHome1/lib32/ldflags    cat
 /oracle/OraHome1/lib32/sysliblist`  -lm  -lpthread -lpthread
/

cd $runs
echo \n\n\t  JXO4CALR COMPLETE
```

For HP-UX and Sun Solaris, procob32 is executed. For AIX RS6000, procob or procob32 is executed.

2.   Execute jxo4calr

To extract COBOL program o4calc from CYBMST, compile the program, and link the machine-specific subroutines. Execute the jxo4calr script from the $runs subdirectory. For example:

rj jxo4calr

Review the log, then the o4calc.03 list file in the $list subdirectory to determine if there were any errors.

Compile and link the delivered cbsvb

1.   Add Oracle linkage before running this script

This delivered script includes a sample ORACLE linkage step (underlined in the script). The linkage step must be replaced by the linkage step created by executing the ORACLE make command.

```
echo \n\n\t  JCMPCVBR IN PROGRESS \n
echo \n\n\t  CHECKING FOR FILES \n
test -f $prog/cbsvb.pco  || echo \n\t cbsvb.pco does not exist
cd $prog
echo \n\n\t COMPILATION OF CBSVB IN PROGRESS \n
procob32  iname=cbsvb.pco oname=cbsvb.cob ireclen=132 oreclen=132 select_error=no
litdelim=apost mode=ansi
cob -xv -C IBMCOMP NOOSVS NOBOUND NOTRUNC ALIGN(8) -N NOCHECK NOBOUNDOPT -o
${file} ${file}.cob rdbpgma.o rdbpgmb.o rdbpgmc.o rdbpgmd.o rdbpgme.o
 rdbpgmf.o rdbpgmg.o rdbpgmh.o \
\
-L/oracle/OraHome1/lib32/ /oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh cat
/oracle/OraHome1/lib32/ldflags    cat
 /oracle/OraHome1/lib32/sysliblist`  -lm  -lpthread -lpthread

/
cd $runs
echo \n\n\t JCMPCVBR COMPLETE
```

2.   Execute jcmpcvbr

To compile the relational batch program CBSVB as delivered, execute the jcmpcvbr script from the subdirectory. For example:

rj jcmpcvbr

Review the log to determine if there were any errors.

Create System Control Repository

Script used: jdemo01r

This procedure creates the System Control Repository. Execute the jdemo01r script. The userid and password parameters should be replaced with the database userid and password. For example:

jdemo01r userid password | tee $log/jdemo01.log

Note: *In the above example the results are piped to a log file.*

Review the log, then the demos01r.03 list file in the `$list` subdirectory to determine if there were any errors. If you receive any 'RELOAD NOT FOUND' messages, like those following, disregard these messages.

```
HEWITT ASSOCS  - RELOAD PROGRAM CHECK
-----------------------------------
P CYBADT ******** RELOAD NOT FOUND ********
P CYBHL   ******** RELOAD NOT FOUND ********
P CYBP15 ******** RELOAD NOT FOUND ********
P CYBPZQ ******** RELOAD NOT FOUND ********
P CYBRCI ******** RELOAD NOT FOUND ********
P CYBSCK ******** RELOAD NOT FOUND ********
P CYBSEC ******** RELOAD NOT FOUND ********
P CYBWCI ******** RELOAD NOT FOUND ********
P CYBWRK ******** RELOAD NOT FOUND ********
P CYBWZQ ******** RELOAD NOT FOUND ********
P CYBX02 ******** RELOAD NOT FOUND ********
P EXCTRL ******** RELOAD NOT FOUND ********
P QMCTRL ******** RELOAD NOT FOUND ********
P RDEMRL ******** RELOAD NOT FOUND ********
P RDEMRQ ******** RELOAD NOT FOUND ********
P RDTBPL ******** RELOAD NOT FOUND ********
P RTCTRL ******** RELOAD NOT FOUND ********
P TBLENT ******** RELOAD NOT FOUND ********
P TBLINQ ******** RELOAD NOT FOUND ********
P TBLUPD ******** RELOAD NOT FOUND ********
P TBLVER ******** RELOAD NOT FOUND ********
P V-NAME ******** RELOAD NOT FOUND ********
P WPTM   ******** RELOAD NOT FOUND ********
```

Note: *This script may run for a while.*

Phase 4: Compile CBSV and CYBIO

Extract all cbsv programs

1. Edit cbsv.ovr

Edit the cbsv.ovr file located in the $data subdirectory to include your database connect string, as in the example shown here:

```
C141250+R -E  01  USERID                       PIC X(08) VALUE CYBORG52
C141300+R -E  01  PASSWD                       PIC X(08) VALUE CYBDBA
C142100+R     01  DB-NAME                      PIC X(8) VALUE CYBORG52
```

2. Execute jpulcvs

To extract all The Administrative Solution CBSV COBOL programs (cbsvo, cbsvot, cbsvb, cbsvbt), execute the jpulcvs script from the $runs subdirectory. The userid and password parameters should be replaced with the database userid and password. For example:

jpulcvs userid password | tee $log/jpulcvs.log

Note: *In the above example the results are piped to a log file.*

Review the log to determine if there were any errors.

Compile and link CBSV programs

1. Add Oracle linkage before running this script

This delivered script includes a sample ORACLE linkage step (underlined in the script). This linkage step must be replaced by the linkage step created by executing the ORACLE make command. For example:

```
echo \n\n\t  JCMPCVR IN PROGRESS \n
echo \n\n\t  CHECKING FOR FILES \n
test -f $prog/cbsvb.pco  || echo \n\t cbsvb.pco does not exist
test -f $prog/cbsvbt.pco || echo \n\t cbsvbt.pco does not exist
test -f $prog/cbsvo.pco  || echo \n\t cbsvo.pco does not exist
test -f $prog/cbsvot.pco || echo \n\t cbsvot.pco does not exist
cd $prog
for file in cbsvb cbsvbt
do
echo \n\n\t COMPILATION OF ${file} IN PROGRESS \n
procob32  iname=${file}.pco oname=${file}.cob ireclen=132 oreclen=132
select_error=no litdelim=apost mode=ansi
cob -xv -C "IBMCOMP NOOSVS NOBOUND NOTRUNC ALIGN(8)" -N "NOCHECK NOBOUNDOPT" -o
${file} ${file}.cob rdbpgma.o rdbpgmb.o rdbpgmc.o rdbpgmd.o rdbpgme.o
 rdbpgmf.o rdbpgmg.o rdbpgmh.o \
\
-L/oracle/OraHome1/lib32/ /oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh cat
/oracle/OraHome1/lib32/ldflags   cat
 /oracle/OraHome1/lib32/sysliblist`  -lm  -lpthread -lpthread
/
echo \n\t COMPILATION OF ${file} COMPLETE \n
done
for file in cbsvo cbsvot
do
echo \n\n\t COMPILATION OF ${file} IN PROGRESS \n
```

```
procob32  iname=${file}.pco oname=${file}.cob ireclen=132 oreclen=132
select_error=no litdelim=apost mode=ansi
cob -xv -C "IBMCOMP NOOSVS -L/oracle/OraHome1/lib32/
/oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh cat
/oracle/OraHome1/lib32/ldflags   cat
 /oracle/OraHome1/lib32/sysliblist  -lm -lpthread -lpthread
NOBOUND NOTRUNC ALIGN(8) -N NOCHECK NOBOUNDOPT -o ${file} ${file}.cob o4calc.o
rdbpgma.o rdbpgmb.o rdbpgmc.o rdbpgmd.o
 rdbpgme.o rdbpgmf.o rdbpgmg.o rdbpgmh.o \
\
-L/oracle/OraHome1/lib32/ /oracle/OraHome1/precomp/lib32/cobsqlintf.o -lclntsh cat
/oracle/OraHome1/lib32/ldflags   cat
 /oracle/OraHome1/lib32/sysliblist`  -lm  -lpthread -lpthread
/
echo \n\n\t COMPILATION OF ${file} COMPLETE \n
done
cd $runs
echo \n\n\t JCMPCVR COMPLETE
```

2.   Execute jcmpcvr

For example:

rj jcmpcvr

Review the log, to determine if there were any errors.

Compile and link CYBIO programs

Script used: jcmpcbio

To compile and link the CYBIO programs, execute the jcmpcbio script from the `$runs` subdirectory. For example:

rj jcmpcbio

Review the log to determine if there were any errors.

Phase 5: Create Test P20IN Batch Master

Extract Report Generators

Script used: jp20strt (U.S.)
jp20strc (Canada)

To extract the report generators from CYBMST and create the P20IN Batch Master File, execute the jp20strt script from the $runs subdirectory with p9strt and p9cbsv as input.

Input files

p05t80	p9strt
p05t81	p9cbsv (U.S.) p9cbsvc (Canada)

The P2EDIT, P4CALC, and P5PRNT programs are processed. For example:

rj jp20strt

Review the log and then all the output listings in the $list subdirectory to determine if there were any errors.

Output listings

audit2.trl
p20strt.03
transload

Create Test Employee Database

Script used: jpaymrg

To create a test Employee Database, execute the jpaymrg script from the $runs subdirectory. For example:

rj jpaymrg

Review the log and then the paymrg.03 list file in the $list subdirectory to determine if there were any errors.

Phase 6: Create Test Employee Database

Populate option list and application tables

Script Used: jpopf01

To synchronize the System Control Repository and relational tables for option lists and specific application tables (Cx through Xx), execute the jpopf01 script from the $runs subdirectory. For example:

rj jpopf01

Populate database with test data

To populate the database with test data, perform the following steps:

1.   Load test companies
Script used: jprdemo (U.S.)
                       jprdemoc (Canada)

This procedure loads the payroll test data into the Employee Database (Master File: FILE02). Execute the jprdemo script from the $runs subdirectory. For example:

rj jprdemo

Review the log and then the prdemo.03 list file in the $list subdirectory to determine if there were any errors.

2.   Load HR test data
Script used: jhrdemo (U.S.)
                       jhrdemoc (Canada)

This procedure loads HR test data into the Employee Database (Master File: FILE02). Execute the jhrdemo script from the $runs subdirectory. For example:

rj jhrdemo

Review the log and then the hrdemo.03 list file in the $list subdirectory to determine if there were any errors.

3.   Build alternate keys
Script used: jbldaky

To build or rebuild the Employee Name Alternate Key, run the jbldaky script located in the $runs directory.

Note: This script may be altered to accommodate your build of the alternate keys once your data has been added to the system.

For example:

rj jbldaky

Review the log and then the bldaky.03 list file in the $list subdirectory to determine if there were any errors.

Extract full payroll data

Script used: jpayxtr

To update the P20IN Batch Master File with the demo test data, execute the jpayxtr script from the $runs subdirectory. The organization and employee data will be extracted from the online Employee Database and placed in FILE12. This will be the new p20in master file.

FILE10 (payxtr10), which contains time entries and adjustments, is also created. payxtr10 becomes input to the jpayrun as p05t81. For example:

rj jpayxtr

Review the log and then the payxtr.03 list file in the $list subdirectory to determine if there were any errors.

Complete a payroll run

Script used: jpayrun

Complete a payroll run to apply taxes (TAXFILE) and the time entries and adjustments (payxtr10) to the p20in file. Execute the jpayrun script from the $runs subdirectory with p05t80 and p05t81 as input. The P2EDIT, P4CALC, and P5PRNT programs will be processed.

Note: *Verify that the TAXFILE being applied is the current tax related regulatory bulletin from CUBBS (Cyborg Users Bulletin Board).*

☞   *Refer to 'Accessing CUBBS ' in the KnowledgeBase or Technical Administration guide for detailed instructions for logging onto CUBBS.*

Input files

p05t80	TAXFILE (US)
	TAXFILEC (Canada)
p05t81	payxtr10

Note: *If you are a Canadian customer, modify the script to extract TAXFILEC instead of the TAXFILE.*

For example:

rj jpayrun

Review the log, then all the output listings in the $list subdirectory to determine if there were any errors.

Output listings

auditrl1.lis
checknum.lis
payslips.lis
depslips.lis
combreg.lis
transload.lis

Perform a payroll maintenance run

Script used: jmntrun

Perform a payroll maintenance run to create pay history and labor records and apply check numbers to the newly created history records on the P20IN Batch Master File. Execute the jmntrun script from the $runs subdirectory. The P2EDIT, P4CALC, AND P5PRNT programs will be processed. For example:

rj jmntrun

Review the log, then the transload2.lis and auditr12.lis list files in the $list subdirectory to determine if there were any errors.

Update the Employee Database

Script used: jpaymrg

To create a new random Employee Database, execute the jpaymrg script from the $runs subdirectory. For example:

rj jpaymrg

Review the log, then the paymrg.03 list file in the $list subdirectory to determine if there were any errors.

Phase 7: Extract HR Reports

Extract HR Reports

Script Used: jreport

To extract Human Resource batch reports, execute the jreport script from the `$runs` subdirectory. For example:

rj jreport

Review the log, then the rtprnt.03 list file in the `$list` subdirectory to determine if there were any errors.

Phase 8: Apply System Control Repository Menu Additions

Build report menus

Script used: jrptmnu

To build a menu containing the delivered report groups, run jrptmnu located in the $runs directory. For example:

rj jrptmnu

Review the log, then the rptmnu.03 list file in the $list subdirectory to determine if there were any errors.

Apply menu and checklist updates for Document Launcher and Enhanced Payroll Reporting

Script used: jmainti

To apply menu items for online user access to batch processes, run the jmainti script from the $runs directory, using the $data/eprddi05 file as FILE05 input. For example:

rj jmainti

Review the log, then the mainti.03 list file in the $data subdirectory to determine if there were any errors.

C H A P T E R   4

Cyborg Application Service (CAS) Installation and Configuration

In This Chapter

Installing and Configuring CAS

This section provides detailed instructions for installing the Cyborg Application Server (CAS) on The Solution Series Application Server on a UNIX system. This is a technical section aimed at system administrators.

Deliverables

The following is included:

1	CD-ROM labeled The Solution Series 5.2 for UNIX
1	Installing and Configuring The Solution Series 5.2 (UNIX) (this guide)

☞   *Refer to Directory Contents for detailed information on scripts used and programs installed during the installation and the purposes they serve.*

Install and Configure the Cyborg Application Server (CAS)

The Cyborg user profile and the CAS script must include the updated values for the ServerExpress environment variables and $path. The CAS sends $data, $work, $list, $prog and $runs to the running CBSVO using the path defined in the cybenv.cfg.

First time install of the Cyborg CAS daemon

Perform the following steps to install the CAS daemon:

1. Verify that CAS can execute on the system

Execute the following command from the directory where you chose to install the CAS script:

```
./cybservd -v
```

CAS should respond with:

```
Cyborg Application Server version x.xx for <system name>
```

Note: *Operating system names are Solaris for Sun Solaris (from Sun Microsystems), AIX for AIX RS6000 (from IBM), and HP-UX for HP-UX (from Hewlett-Packard).*

The CAS script must be restarted to register these changes. If a message such as 'Exec format error' appears, then CAS will not run on the system, and CAS needs to be compiled specially for that particular UNIX server. If an error appears and CAS will not run, please contact Customer Support.

2. Edit and add environment variables to CAS script in the MicroFocus/Oracle section.

These are some of the same environment variables defined in the Cyborg profile.

For AIX RS6000

```
COBDIR=/opt/lib/cobolsp1                      ; export COBDIR
LIBPATH=$COBDIR/lib:$LIBPATH                        ; export LIBPATH
ORACLE HOME=                                  ; export ORACLE_HOME
ORACLE PATH=                                  ; export
ORACLE_PATHNLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252; export NLS_LANG
```

For Sun Solaris

```
COBDIR=/opt/lib/cobolsp1                      ; export COBDIR
LD_LIBRARY_PATH=$COBDIR/lib:$LD_LIBRARY_PATH ; export LD_LIBRARY_PATH
ORACLE HOME=                                  ; export ORACLE_HOME
ORACLE PATH=                                  ; export ORACLE_PATH
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252                 ; export NLS_LANG
```

For HP-UX

```
COBDIR=/opt/lib/cobolsp1                      ; export COBDIR
SHLIB_PATH=$COBDIR/lib:$SHLIB_PATH            ; export SHLIB_PATH
ORACLE HOME=                                  ; export ORACLE_HOME
ORACLE PATH=                                  ; export ORACLE_PATH
```

```
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252                    ; export NLS_LANG
```

3.   Add the Cyborg port to the network services file

Because CAS is a TCP/IP server, it uses a system resource called a *port number*. A port number is simply a number at which a server program can be located. Port number 9888 is reserved for CAS. The port will be configured in a system-wide file called `/etc/services`.

1. Edit the file `/etc/services` to add the port to the network services file. The entries in the file are usually given in numeric order, so find the location where 9888 should appear, then add the following line to the file:

```
cyborg     9888/tcp     # Assigned by IANA to Cyborg Systems
```

Note: *You also need to enter the 9888 port number on the Connection Editor during client installation, as described in* **Installing and Configuring the Administrative Client** *(on page 59).*

2. Save the file and exit the editor.

☞   *If you have any problems with CAS installation, see the Troubleshooting section in* **Administering the Cyborg Application Server (CAS) Daemon** *(on page 131).*

4.   Add CAS to the list of programs to launch at system startup

Add the CAS script to the list of programs to launch at startup (cd to CAS directory and execute the script). This ensures that clients will be able to access The Solution Series even if the machine is rebooted.

Subsequent installations of CAS

Use the following instructions if you already have the CAS daemon installed. Since the CAS script includes references to the associated compiler, when a different compiler is used, it is necessary to install a new CAS. Perform the following steps to install another CAS daemon:

1.   Verify that CAS can execute on the system

Execute the following command from the `/cyborghome/app/server` directory:

```
./cybservd -v
```

CAS should respond with:

```
Cyborg Application Server version x.xx for <system name>
```

Note: *Operating system names are Solaris for Sun Solaris (from Sun Microsystems),* `AIX` *for AIX RS6000 (from IBM), and* `HP-UX` *for HP-UX (from Hewlett-Packard).*

The CAS script must be restarted to register these changes. If a message such as 'Exec format error' appears, then CAS will not run on the system, and CAS needs to be compiled specially for that particular UNIX server. If an error appears and CAS will not run, please contact the Help Desk.

2.   Edit and add environment variables to CAS script in the MicroFocus/Oracle section.

These are some of the same environment variables defined in the profile.

For AIX RS6000:

```
COBDIR=/opt/lib/cobolspl                      ;export COBDIR
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252         ;export NLS_LANG
LIBPATH=$COBDIR/lib:$LIBPATH                   ;export LIBPATH
ORACLE HOME=                                   ;export ORACLE_HOME
ORACLE PATH=                                   ;export ORACLE_PATH
```

For Sun Solaris:

```
COBDIR=/opt/lib/cobolspl                      ;export COBDIR
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252         ;export NLS_LANG
LD_LIBRARY_PATH=$COBDIR/lib:$LD_LIBRARY_PATH ;export LD_LIBRARY_PATH
ORACLE HOME=                                   ;export ORACLE_HOME
ORACLE PATH=                                   ;export ORACLE_PATH
```

For HP-UX:

```
COBDIR=/opt/lib/cobolspl                      ;export COBDIR
NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252         ;export NLS_LANG
SHLIB_PATH=$COBDIR/lib:$SHLIB_PATH             ;export SHLIB_PATH
ORACLE HOME=                                   ;export ORACLE_HOME
ORACLE PATH=                                   ;export ORACLE_PATH
```

3.   Add the new port number to the CAS script

Because CAS is a TCP/IP server, it uses a system resource called a *port number*. A port number is simply a number at which a server program can be located. Port number 9888 is reserved for the CAS. If this port is already in use, please select the next available number for use by this CAS. There is no need to edit the file called /etc/services. When using CASMGR, you will need to include reference to this port number. Edit the cas script. On the last line, enter the port number, as follows:

$daemon -p9889 -a$appcfg -e$envcfg -T$tracedir -c$maxconn

Note: *You also need to enter the port number on the Connection Editor during client installation, as described in* **Installing and Configuring the Administrative Client** *(on page 59).*

☞   *If you have any problems with CAS installation, see the Troubleshooting section in* **Administering the Cyborg Application Server (CAS) Daemon** *(on page 131).*

4.   Add CAS to the list of programs to launch at system startup

This step should be performed by the administrator. Add the script:

/cyborghome/app/server/cas

to the list of programs to launch at system startup. This ensures that clients will be able to access The Solution Series even if the machine is rebooted.

Start CAS

CAS can only be started by the administrator or a similarly privileged process. To launch CAS, log in as root and execute the script CAS. The following output should be displayed:

```
CAS version x.xx for <system name> started successfully (process ID xxxx)
```
where xxxx is the process ID of the daemon.

Note: *Operating system names for* Solaris *for Sun Solaris (from Sun Microsystems),* AIX *for AIX RS6000 (from IBM), and* HP-UX *for HP-UX (from Hewlett-Packard).*

The command prompt is returned to the user, as CAS has placed itself into the background and will continue to run in the background. This script encapsulates the settings that are used by this installation.

☞   *For information on the CAS script or are unable to start CAS, see* **Appendix E** *(see "Administering the Cyborg Application Server (CAS) Daemon" on page 131)*

Configure the environment

To configure the environment for this installation, follow the directions in the 'Configuring a new environment on the server' section in *Appendix E* (see "Administering the Cyborg Application Server (CAS) Daemon" on page 131).

P ART 3

Installing and Configuring the Administrative Client

In This Section

CHAPTER 5

Installing and Configuring the Administrative Client

In This Chapter

Introduction

This chapter provides detailed instructions for configuring clients of The Solution Series. This is a technical chapter aimed at system administrators.

Deliverables

The following is included:

1	CD-ROM labeled The Solution Series 5.2 for Windows
1	Installing and Configuring The Solution Series 5.2 (Windows) (this guide)

☞   *Refer to Directory Contents for detailed information on scripts used and programs installed during the installation and the purposes they serve.*

Phase 1: Prepare for installation

It is important to appropriately prepare for the client installation of The Solution Series. Because the client and server will work together, you must ensure that they are both synchronized in communication.

Before successful operations can be carried on between the server running The Solution Series and the client, you are required to install and configure the prerequisite software described in Chapter 1. This software must be in place before beginning the installation.

Note: *Based upon your employees individual client workstation setups, there may be missing shared .dll or .ocx files that the product uses that are non-distributable Microsoft shared system files. If this occurs, your IT staff will need to locate and download (usually on the Microsoft website) the required file for inclusion on your PCs.*

Ensure Cyborg Application Service is active

The Cyborg Application Service must be running on the server.

☞ *Refer to Install and Configure Cyborg Application Server (CAS) service to learn how you can tell if CAS service is active.*

Complete a configuration worksheet

During the installation, you will be prompted to supply client and server information. Complete the table below in advance preparation for these entries:

	Description	Your Configuration
Installation Location	Identify where you want the client files installed. If you accept the default location, the files will be placed at `C:\Program Files\Cyborg Systems\Clientxx` (where xx indicates the release level of the installation).	
Connection Type	The only connection type available at this time is the Cyborg Application Service (CAS).	
Connection Name	Identify a title for the connection you will configure between the client and the server. For the initial installation, accept the default connection name (Defaultxx, where xx indicates the release level of The Solution Series installation). If a user requires multi-environment access, additional environments can be set up later. This label must not contain the characters \\/:*?\<>_. If you name the connection by another name, additional steps must be performed. Refer to the appendices for more information on creating *Multiple Environments* (see "Creating Separate Environments on the Server for the Client" on page 109).	
Host	Host name or IP address of the server to which you want the client to connect.	
Port	Identify the port address of the server. We registered the port address of 9888 for the Cyborg Application Service (CAS). This address must be the same as was specified during the installation and configuration of Cyborg Application Service (CAS).	9888

	Description	**Your Configuration**
Environment	Identify the environment name (up to 8 characters) that was established during the Cyborg Application Service (CAS) installation, for example, CYBORG, CYBPROD, CYBTEST, or CYBDEV. *Note: The environment name should contain no space characters, and should reflect the name of the Cyborg Application Service (CAS) for that environment.*	

Note: *If installing the client on the Windows server, then either enter localhost or 127.0.0.1.*

Phase 2: Install the software

Install client files

Be sure to have at hand the configuration worksheet completed as preparation as you may need to refer to it as you load programs on the PC.

1.   **Insert the Solution Series CD-ROM into the Administrative client PC**
The Getting Started page displays.

2.   **Navigate and then click on the Install Administrative client link**
Clicking this link allows you to either launch the autoinstall or save the file to a local PC for later installation.

3.   **Follow the installation prompts**
The following table tells you what information the installation program will require. If you need anything other than the defaults, use the information in the Select the following... column.

Prompt	Options/Defaults	Select the following…
Destination Folder	C:\Program Files	Use default **or** Other:
Are you running The Solution Series on an zOS Server?	--	No
Launch the Connection Editor?	Selected (Yes)	Deselect this option if you are installing Document Management Facility and/or Enhanced Payroll and Reporting (EPR) (Windows and UNIX only). Select this option if you are **not** installing Document Management Facility and/or Enhanced Payroll and Reporting (EPR) and are ready to configure the client connection.

The installation program will prompt you when it is complete.

(Optional) Install Enhanced Payroll and Reporting

If you wish to use Enhanced Payroll and Reporting, you must first install the EPR client files.

1. **Insert the Solution Series CD-ROM into the Administrative Client PC**
   The getting Started page displays.

2. **Navigate to and click Install Enhanced Payroll and Reporting link**
   Clicking this link allows you to either launch the autoinstall or save the file to a local PC for later installation.

3. **Follow the installation prompts**
   The following table tells you the information required by the installation program. If you need anything other than the defaults, use the information in the Select the following... column.

Prompt	Options/Default	Select the following...
Destination Folder	C:\Hewitt\Client52	Use default **or** Other:

The installation program will prompt you when it is complete.

Note: *Prior to using the Process Monitor, you must specify the correct environment and user folder. The Specify Environment Folder and Specify User Folder dialogs will display the first time the Process monitor utility is run.*

This requires you to map a windows drive to the /users directory in the applicable environment(s) on the UNIX Server. This requires either NFS or SAMBA Software to be installed and configured. NFS and SAMBA are 3rd party tools and are outside the scope of this guide.

(Optional) Install Document Management Facility

If you want to use the Document Data Interface, you must first install the Document Management Facility client files.

1. **Insert the Solution Series CD-ROM into the Administrative Client PC**
   The Getting Started page displays.

2. **Navigate to and click on the Install Document Management Facility link**
   Clicking this link allows you to either launch the autoinstall or save the file to a local PC for later installation.

3. **Follow the installation prompts**
   The table following tells you what information the installation program will require. If you need anything other than the defaults, use the information in the Select the following… column.

Prompt	Options/Defaults	Select the following…
Destination Folder	C:\Hewitt\Client52	Use default **or** Other:

The installation program will prompt you when it is complete.

Phase 3: Configure the software

Set Up Your Environment

To set up your environment, perform the following steps:

1.   Access the Connection Editor

Access this dialog box at the end of the installation by selecting:

Start ▶ All Programs ▶ Hewitt Associates ▶ The Solution Series Admin v52 ▶
Connection Editor

2.   Click New

Click on the New button to set up a new configuration.
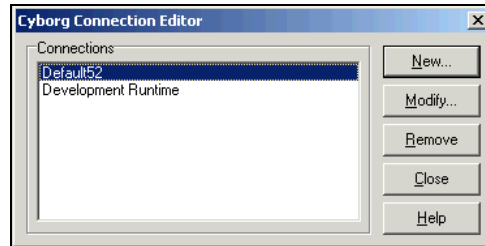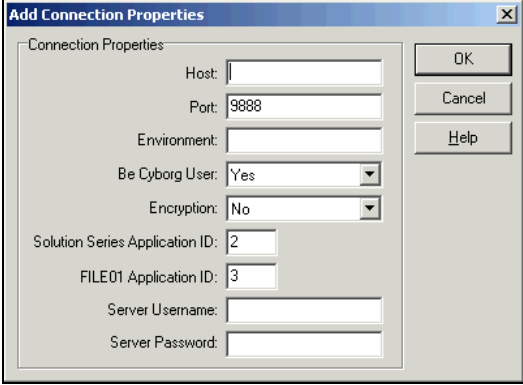
3.   Type the name of your new connection

Leave Cyborg Application Service selected in the Type drop-down list box. Type the name
of the connection you are creating in the Name text box.
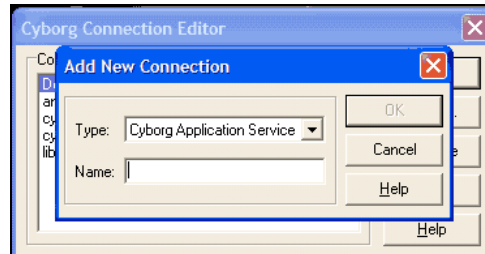
4.   Click OK

The Add Connection Properties dialog displays.

5.   Type the configuration details

Type the configuration details under the Connection Properties on the Add Connection
Properties dialog box.

Item	Description
Host	Identify the system name of the server.
Port	Identify the port address of the server. The port address of 9888 has been registered for the Cyborg Application Server (CAS). Be sure to use the same port on the server.
Environment	Identify the environment name (up to 8 characters) on the server. Examples are: CYBPROD, CYBTEST, and so on. (See the environment name on your configuration worksheet.)
Be Cyborg User	Accept the default entry of Yes to enable the Cyborg User.
Encryption	The default entry is No. If Yes is entered, the sign-on data sent between the client and the server will be encrypted.
Solution Series Application ID	The application name and ID number for The Solution Series application. This application ID is configured to 2.
FILE01 Application ID	The application name and ID number for the CYBIO application. This application ID is configured to 3.
Server Username	Leave this field blank.
Server Password	Leave this field blank.

6. **Click OK**

The connection properties are specified.

7. **Click Close**

The connection has been configured between the server and the client.

Phase 4: Test the installation

Test the connection to the server

To test the connection to the server, perform the following steps:

1. Launch The Solution Series
Select:

Start ▶ All Programs ▶ Hewitt Associates ▶ The Solution Series Admin v5.2 ▶ The Solution Series Administrative Client v5.2
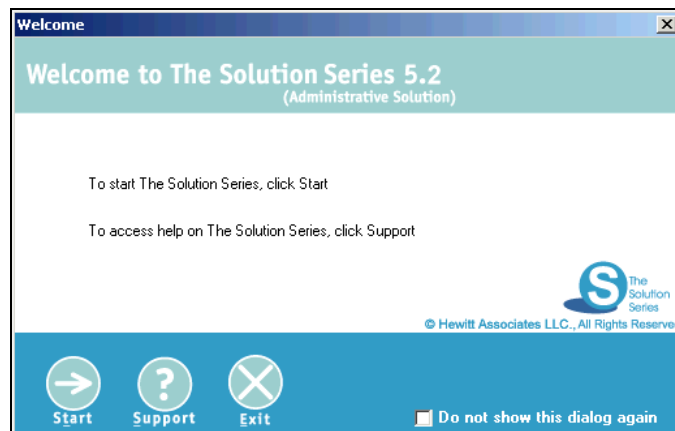
2. Click Start on the Welcome screen
The Login dialog box appears.

Note: *If you see an error message instead of this Welcome screen, refer to the error listing in Optional SQL Server Procedures or Optional ORACLE Procedures for further information.*

Test the GUI

To test the GUI, perform the following steps:

1.   Log on as Security Officer
Select the environment you want to access from the option list and then type your user name and password:
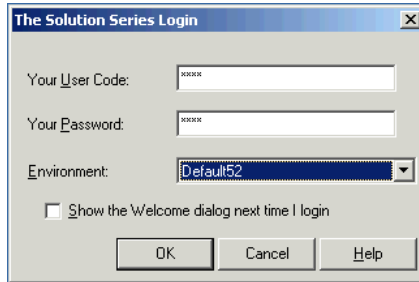
2.   Click OK
If FILECL32, the Client Data File, does not exist on your machine, the following dialog displays:
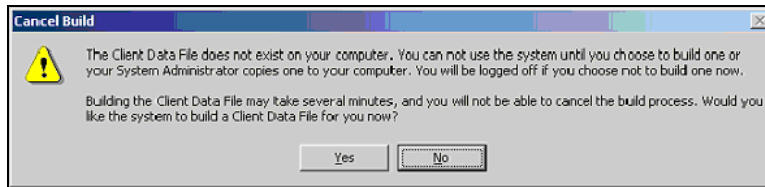
3.   Click Yes.
The Solution Series builds the Client Data File and prompts you to build the fields and option lists.
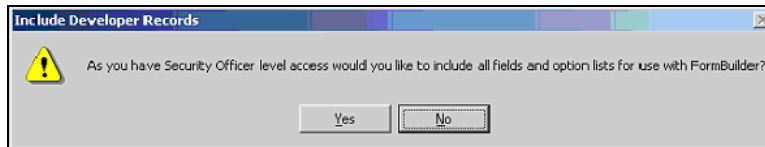
4.   Click Yes
The Solution Series builds the fields and option lists, and the work area for The Solution Series displays.

☞   *Refer to Creating Separate Environments for information on creating additional environments.*

View the Favorites Toolbar

1.   Logon

2.   To add the Launch Bar to the Menu Bar, customize the toolbar
<u>V</u>iew ▶ Customize Toolbars

On the Toolbars tab, click beside Launch.
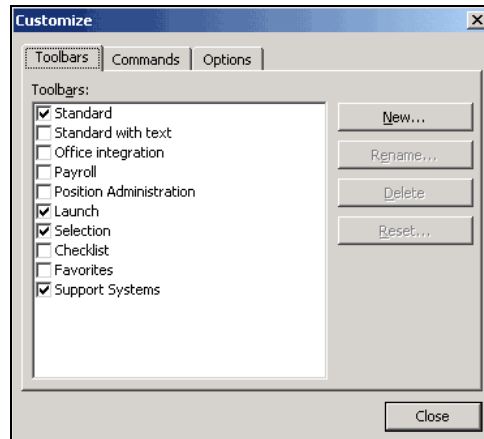
3.   Click Close

The Launch toolbar appears.

Define the email and letter template folder

Document templates are created when a letter or email communication event is set up. These templates contain the body of the letter or email. Where the templates are stored is important. The location is specified by the Main Document Path text box on the System Options form (SCOPTS).

Before setting the Main Document Path, you need to create or determine which folder will be used for storing the templates. Communication events that will be used by multiple users should be stored on a network drive. This allows any user who might trigger the event, either manually or automatically through an action or condition, to access the templates.

Note: You perform this task only once.

To set the Main Document Path, complete the following steps:

1.   Access the System Options form (SCOPTS)

Access this form by selecting the following:

Component:		Security Tools
Process:		Security Tools
Task:		Specify System Options

The System Options form (SCOPTS) appears.

Note: In the Sign-on Default field, enter a valid control 1-2 for your organization.

2.   Enter the Main Document Path

In the Main Document Path box, type in the path of the folder that contains the email and letter templates that the Office Integration will utilize. For general use, it is required that this path be accessible to all Administrative Clients.

3.   Press Enter

The Main Document Path has now been set allowing the system to find the path where the templates are stored.
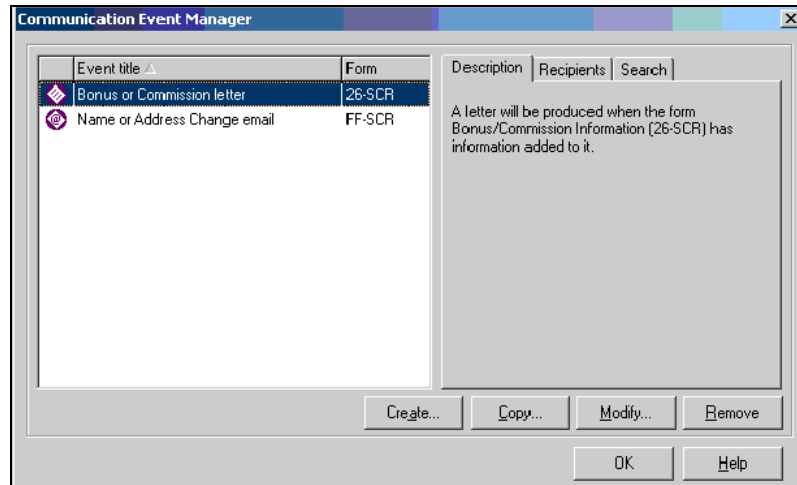
4.   Exit The Solution Series

Before the Main Document Path will take effect, you need to log off and log back onto The Solution Series.

Test Word integration
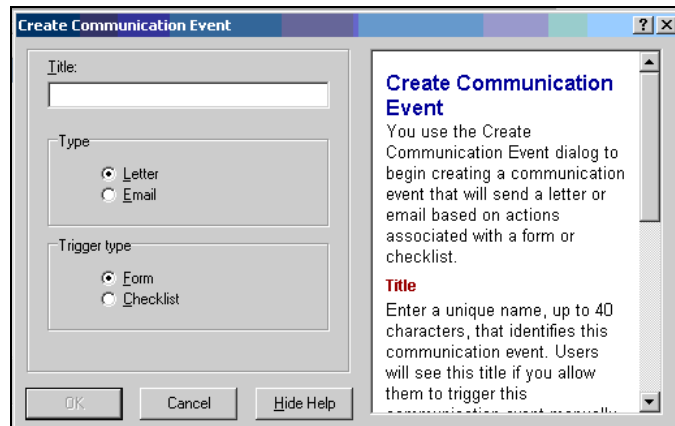
1.   Access the Communication Event dialog

Access this dialog by making the following selections from the Navigator:

Component:	🔧	Administrator Tools
Process:		Administrator Tools
Task:	💬	Manage Events

2.   Click Create

The Create Communication Event dialog appears:

3.   Enter a Title

Type a unique title, up to 40 characters, for this communication event. For the test letter, type the following:

Test Letter

4.   Select Letter

By default, Letter is the communication event Type selected when the dialog is first displayed. Verify that Letter is selected.

5.   Select Form

By default, Form is the communication event Trigger type selected when the dialog is first displayed. Verify that Form is selected.

6. Click OK

Click OK to create the communication event. The Modify Communication Event dialog appears:

7. Select the Form name

Select the form on which this communication event will be triggered by clicking the ellipsis (…) on the right side of the Form name list box. The Browse Forms dialog will show the menu structure from which you can select a form by drilling down through the menus. For the test letter, select:

Employee Payroll ► Add and Maintain Employee Details ► Personal Information

8.   **Click OK**

This will select the form.

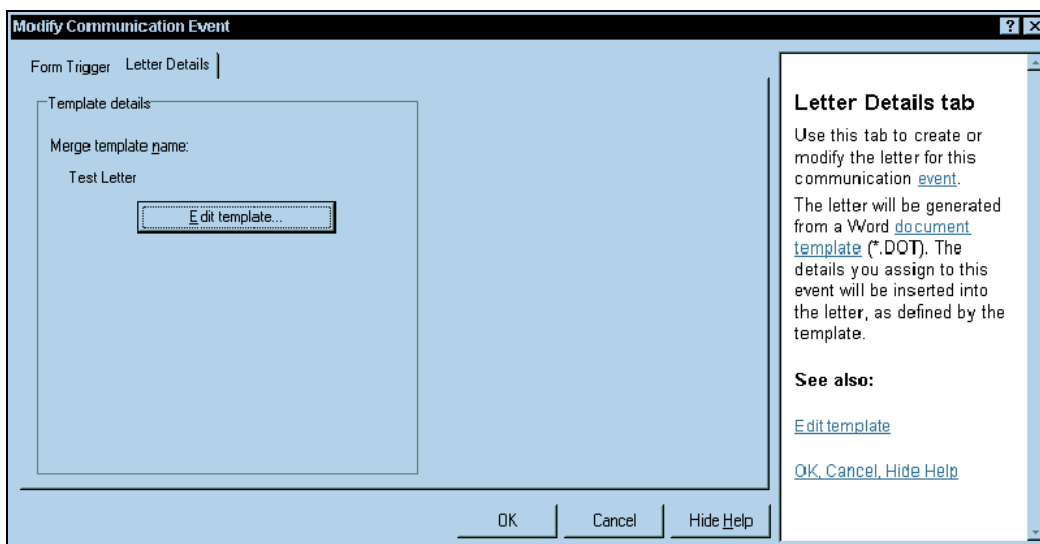**9.**   **Select the automatic trigger conditions**

Select Trigger automatically. This will activate the trigger options.

10.   **Select When information is amended**

When information is added is the default selection—change this.

11.   **Click on the Letter Details tab**

The Letter Details tab appears:

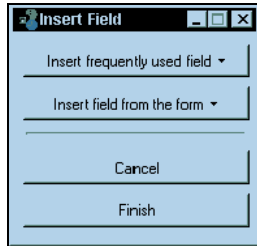**12.**   **Click Edit Template**

This will open Word and the New dialog. The New dialog allows you to select existing templates on which to base the new one.

13.   **Select Blank Document**

This is the default.

14. Click OK

Word creates a new blank document, and the Insert Field dialog appears:

The Insert Field dialog allows you to include information directly from The Solution Series in the template.

15. Add the First Name field

Click Insert frequently used field' on the Insert field dialog, and then select First Name from the drop-down list. This selection and the next couple tests whether The Solution Series information is being placed in the Word template properly.

16. Type a space in the work area

17. Add the Last Name field

Click Insert Frequently used field on the Insert Field dialog and then select Last Name from the drop-down list.

18. Press Enter twice in the work area

This will add two lines to the Word document.

19. Type in a line in the work area

For the test letter, type the following:

has moved to the following address:

20. Press Enter twice in the work area

This will add two lines to the Word document.

21. Add the Address_Line_1 field

Click Insert frequently used field on the Insert field dialog and then select Address_Line_1 From the drop-down list. The display should appear like this:

22. Click Finish

Word will save the letter.

23. Click OK

This will close the Modify Communication Event dialog.

24. Click OK

This will close the Communication Event Manager dialog.

25. Access the Name and Address form

Make the following selections:

Component:		Employee Resourcing
Process:		Maintain Employee Details
Task:		Basic Employee Information

26. Select an employee

In the Number field, type: 0003 in the Control 1-2 of 777777.
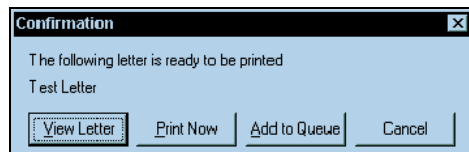
27. Click OK

The Employee Information form (EF-SCR) appears with the employee information:

28. Type a new address

In the Address Line 1 field, type the following: 17 Moore House.

29. Press Enter

This will enter the new address. At this point, the Confirmation dialog appears:

30. Click View Letter

The system opens up the letter in Word and it includes the employee information.

31. Close the application

Review the letter and close the application.

Test email integration

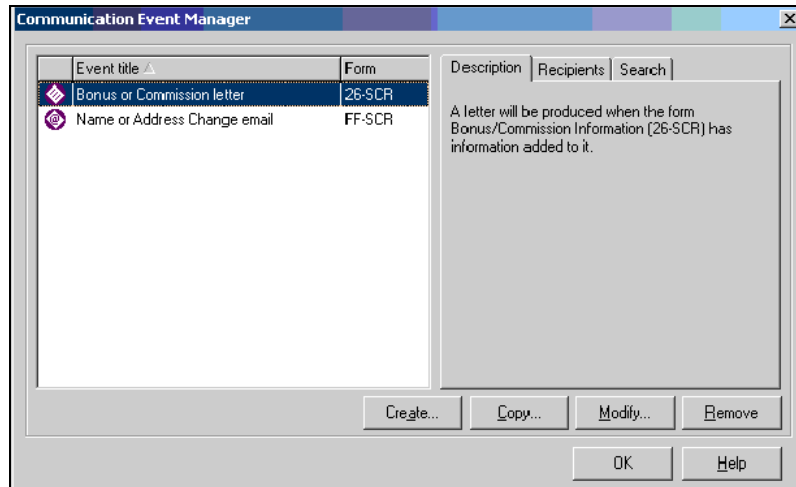**1. Access the Communication Event dialog**

Access this dialog by making the following selections from the Navigator:

Component:		Administrator Tools
Process:		Administrator Tools
Task:		Manage Events

Note: *At this point, you may want to delete the Test Letter created in the task Test Word integration. This can be done on the Communication Event dialog by selecting Test Letter in the Event list, then clicking Remove.*

2.   Click Cre<u>a</u>te

The Create Communication Event dialog appears:
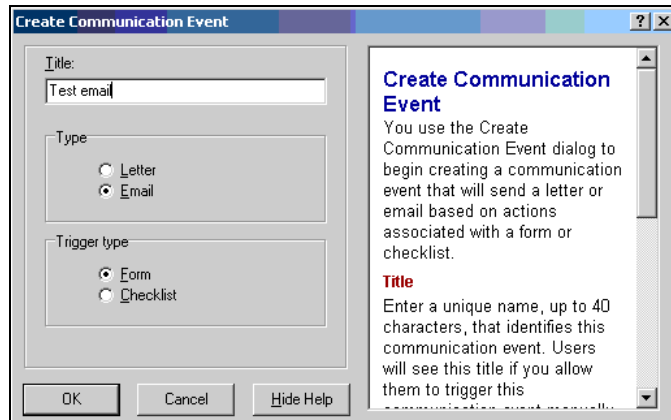
3.   Enter a <u>T</u>itle

Type a unique title, up to 40 characters, for this communication event. For the test email type the following:

Test Email

4.   **Select _E_mail**
By default, _L_etter is the communication event Type selected when the dialog is first displayed. Change the selection to _E_mail.

5.   **Select _F_orm**
By default, _F_orm is the communication event Trigger type selected when the dialog is first displayed. Verify that _F_orm is selected.

6.   **Click OK**
The Modify Communication Event dialog appears:

7.   **Select the _F_orm name**
Select the form on which this communication event will be triggered by clicking the ellipsis (…) on the right side of the _F_orm name list box. The Browse Forms dialog will show the menu structure from which you can select a form by drilling down through the menus.

For the test email, select:

Employee Payroll ▶ Add and Maintain Employee Details ▶ Personal Information
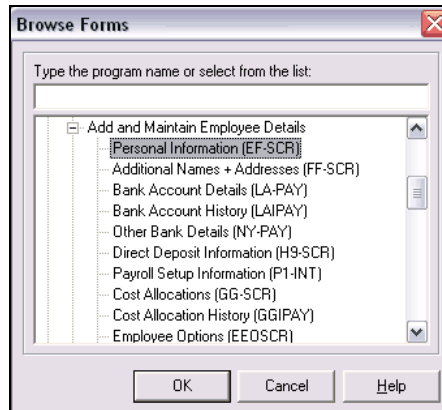
8.   **Click OK**

9.   **Select the automatic trigger conditions**
Select Trigger automatically. This will activate the trigger options.

10.   **Select When information is amended**
When information is added is the default selection—change this.

11.   **Click the Email Details tab**
The Email Details tab will appear.

12. In the Employee field, select To:

Click on the down arrow, and a drop-down list will appear. Select To: from the list. Once you select To, the system automatically includes the Employee field, in order to obtain the email address of the required employee.

13. Select another email address

Click Address book lookup and select the desired email address. This will send the email to another address, which you can use to check that the email has been received. You may want to use your own or another easily accessible address.

14. Click OK

This will select the email address and return you to the Modify Communication Event dialog.

15. Click on the Display email preview option

This is located in the lower left corner of the Email Details tab. Once this option is selected, the system will automatically generate a dialog that prompts you when it is generating the email.

16. **Click on the Email Text tab**

The Email Text tab appears:

This is where you create the email.

17. **Type the Subject**

In the subject field, type the following:

Test Email

18. **Add the First Name field**

Click Insert frequently used field on the Insert field dialog and then select First Name from the drop-down list. This selection and the next couple will test if The Solution Series information is being placed in the Word template properly.

19. **Type a space**

20. **Add the Last Name field**

Click Insert frequently used field on the Insert Field dialog and then select Last Name from the drop-down list, and click in the work area.

21. **Press Enter twice**

This will add two lines to the email.

22. **Type in a line**

For the test letter, type the following line:

has moved to the following address:

23.   Press Enter twice

This will add two lines to the email.

24.   Add the Address Line 1 field

Click Insert frequently used field and then select Address Line 1 from the drop-down list. The display should appear like this:

25.   Click OK

This will enter the data and return you to the Communication Event Manager dialog.

26.   Click OK

This will close the Event Manager dialog. The Employee Personal Information form (EF-SCR) displays with the employee information.

27.   Type a new address

In the Address Line 1 field, type the following: 19 Moore House.

28.   Press Enter

Press Enter to update the new address. At this point, the Email Preview dialog appears.

29.   Click Send

This will send the email to the selected address. When you are finished with this task, you should check the In Box at the address the email has been sent to in order to verify that it has been received. It is recommended that you use an email address you can easily access.

Test the import facility

Standard test data must be loaded. This task will walk you through a test import with a sample Excel spreadsheet in order to ensure that the import functionality is working properly.

1.   Access the Import Profile Manager dialog

Access this dialog by selecting the Import Manager icon from the toolbar:

Alternatively, make the following selections from the menu:

Actions ▶ Office Integration ▶ Import

The Import Profile Manager dialog is displayed:
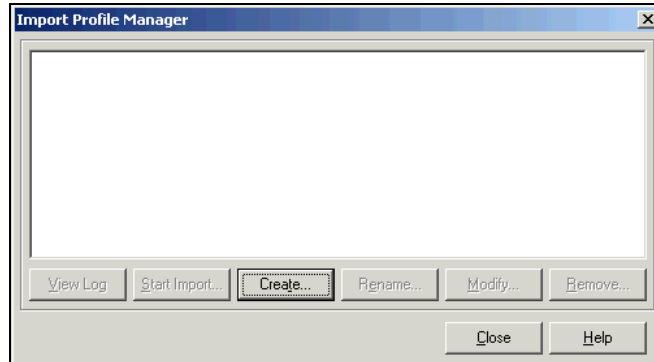
2.   Click Create

Click Create to activate the Import Creation and Amendment wizard.

3.   Click Next

4.   Click Browse

If the form displayed in the dialog is not the form to be selected for import, or if no form is being displayed, click Browse, and browse to the correct directory and file to be imported. Forms that cannot be used with the import functionality do not display.

Use this dialog to select The Solution Series form into which the test file will be imported.

5.   Select The Solution Series form

For the test file, make the following selections:

Employee Resourcing ▶ Maintain Employee Details ▶ Vehicle information
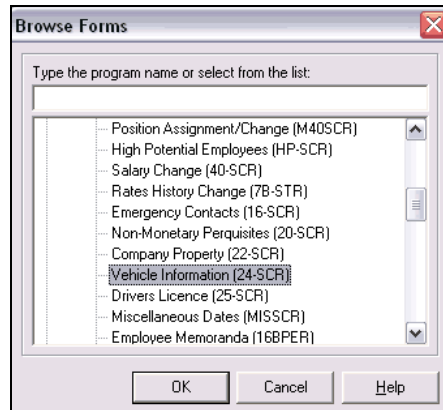
6.   **Click OK**

   Clicking OK will select The Solution Series form.

7.   **Click Next**

8.   **Click Browse**

   The Open dialog will appear. Use this dialog to select the import file.

9.   **Select the import file**

   Use the Open dialog to find the following file path:

   \Program Files\AdminSolutions\Clientxx\Samples\car.xls

   Where xx is the version number.

10.   **Click Open**

   This will select the Excel file.

11.   **Click Next**

12.   **Select the First record is a header option**

   This option tells the system to use the first record in the spreadsheet as a column heading.

13.   **Click Next**

   Now you must establish the relationship between the spreadsheet you are importing and the form into which you exporting it.

14.   **Click Next**

   Define the relationship between the spreadsheet and the form. The next step of this task will explain more about this relationship.

15. Map the import-to-form relationship

Use the select the name method to map the fields in the spreadsheet to the Automobile Information form.

- For each spreadsheet field displayed in the bottom section of the dialog, click on the top row of the column. A drop-down list will display.
- Choose the field name from the drop-down list that matches the column names. The top row of the column will be updated to show the field name, and the matching field on the form will change to yellow to show that mapping has occurred. You will not see any color changes for the fields organization and employee.

- Type the letter T in the Date field. This causes the current date to be used. The field color will change to blue.
- Leave the other fields on the form blank.

16. Click Next

17. Click Finish

The Wizard will return you to the Import Profile Manager, and the import profile you have just created will appear in the dialog.

18. Select the desired import

Select the Import from car to Automobile Information form.

19. **Select <u>S</u>tart Import**

    Click <u>S</u>tart Import to start the import of data to The Solution Series. The Import Options dialog displays:
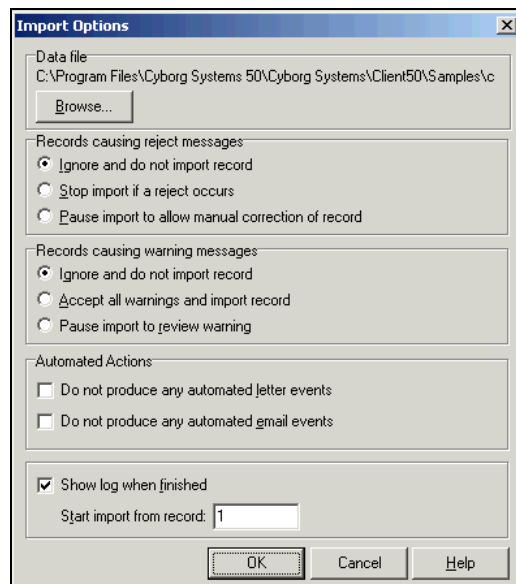
20. **Click OK**

    Click OK to continue the import. The Solution Series will display a log after the import is complete. If no errors are reported, then the import was a success.

21. **Review and close the log**

22. **Close the Import Dialog Box**

Your installation of The Solution Series for Microsoft Windows on the Administrative Client is now complete.

P ART 4

Appendices

In This Section

A P P E N D I X A

Installation Checklists

In This Appendix

Installation Checklist - Indexed Administrative Solution Installation and Configuration

Phase 1: Prepare for installation
- ❑ Install and configure prerequisite software
- ❑ Create user and file system

Phase 2: Extract and Transfer Install Files
- ❑ The Solution Series directory structure
- ❑ Extract install files to a PC
- ❑ Transfer install files from the PC to the server
- ❑ Extract server install files
- ❑ Set up user profile
- ❑ Verify compiler versions
- ❑ Check special requirements

Phase 3: Compile Batch Programs and Build FILE01
- ❑ Extract and compile cybmst programs
- ❑ Compile and link sort programs
- ❑ Compile and link the delivered cbsvb
- ❑ Compile and link the US Quarterly Processor
- ❑ Create System Control Repository

Phase 4: Compile CBSV and CYBIO
- ❑ Extract cbsv programs
- ❑ Compile and link CBSV programs
- ❑ Compile and link CYBIO programs

Phase 5: Create Test P20IN Batch Master
- ❑ Extract Report Generators
- ❑ Create Test Employee Database

Phase 6: Create Test Employee Database
- ❑ Populate database with test data
- ❑ Extract full payroll data

❑ Complete a payroll run

❑ Perform a payroll maintenance run

❑ Update the Employee Database

Phase 7: Extract HR reports

❑ Extract HR Reports

Phase 8: Apply System Control Repository Menu Additions

❑ Build report menus

❑ Apply menu and checklist updates for Document Launcher and Enhanced Payroll Reporting

Installation Checklist - Relational Administrative Solution Installation and Configuration

Phase 1: Prepare for installation

- ❑ Install and configure prerequisite software
- ❑ Create user and file system

Phase 2: Extract and Transfer Install Files

- ❑ The Solution Series directory structure
- ❑ Extract install files to a PC
- ❑ Transfer install files from the PC to the server
- ❑ Extract server install files
- ❑ Set up user profile
- ❑ Verify compiler versions
- ❑ Check special requirements

Phase 3: Build The Solution Series environment

- ❑ Extract and compile all cybmst programs
- ❑ Compile and link sort programs
- ❑ Compile and link the US Quarterly Processor
- ❑ Compile rdbpgm0.cob
- ❑ Execute the CASE tool
- ❑ Execute the make command
- ❑ Pre-compile, compile, and link rdbpgm1
- ❑ Create the tablespaces, tables, indexes, and views
- ❑ Pre-compile, compile, and link rdbpgma through rdbpgmh
- ❑ Extract, compile, and link o4calc
- ❑ Compile and link the delivered CBSVB
- ❑ Create System Control Repository

Phase 4: Compile CBSV and CYBIO

- ❑ Extract all cbsv programs
- ❑ Compile and link all CBSV programs
- ❑ Compile and link CYBIO programs

Phase 5: Create test P20IN Batch Master

❑ Extract Report Generators

❑ Create Test Employee Database

Phase 6: Create Test Employee Database

❑ Populate option list and application tables

❑ Populate database with test data

❑ Extract full payroll data

❑ Complete a payroll run

❑ Perform a payroll maintenance run

❑ Update the Employee Database

Phase 7: Extract HR reports

❑ Extract HR reports

Phase 8: Apply System Control Repository Menu Additions

❑ Build report menus

❑ Apply menu and checklist updates for Document Launcher and Enhanced Payroll Reporting

Installation Checklist - Cyborg Application Service (CAS) Installation and Configuration

Install and Configure the Cyborg Application Server (CAS)

❑ Install CAS

❑ Start CAS

❑ Configure the environment

Installation Checklist - Installing and Configuring the Administrative Client

Phase 1: Prepare for installation

❑ Ensure Cyborg Application Service is Active

❑ Complete a configuration worksheet

Phase 2: Install the software

❑ Install Client files

❑ (Optional) Install Enhanced Payroll and Reporting (EPR)

❑ (Optional) Install Document Management Facility

Phase 3: Configure the software

❑ Set Up Your Environment

Phase 4: Test the installation

❑ Test the connection to the server

❑ Test the GUI

❑ View the Favorites Toolbar

❑ Define the email and letter template folder

❑ Test Word integration

❑ Test email integration

❑ Test the import facility

A P P E N D I X   B

Directory Contents

In This Appendix

Server directory structure

The server software is delivered in two main directories:
`/cyborghome/app` and `/cyborghome`

/cyborghome/app directory

The `/cyborghome/app` directory contains application-specific delivered source and executables.

- The `/cyborghome/app/server` subdirectory contains the delivered scripts and executable of the Cyborg Application Server (CAS) daemon. This directory will also contain the system log of CAS execution and events.

- The `/cyborghome/app/server/trace` subdirectory will contain the detail trace files of execution of every client when CAS is in debug mode.

/cyborghome/cyborgxx directory

The `/cyborghome/cyborgxx` root directory contains the following subdirectories:

Subdirectory	Description
`bacs`	Output for BACS
`data`	Data files and control records
`list`	Output from script processing
`log`	Optional output from script processing
`prog`	Source and executable files
`runs`	Delivered scripts
`updates`	Delivered source files
`users`	Directory created by process monitor
`work`	Temporary files

/cyborghome/app-server directory

The following files are shipped with CAS under the `/cyborghome/app-server` directory:

Filename	Description
cas	Script to start CAS
casadduser.pl	CAS script
casdeluser.pl	CAS script
casmgr	CAS administration utility
caspasswd.conf	CAS script
caspasswd.pl	CAS script
cybservd	The CAS server executable

Filename	Description
cybapp.cfg	List of The Solution Series applications
cybenv.cfg	List of The Solution Series environments
viewlog	Prints all detail from the System Log/trace file
viewlog.msg	Prints only the log message from the System Log/trace file

The following files are created by CAS as it executes:

Filename	Description
/cyborghome/app/server/system.log	Log of CAS execution and events
/cyborghome/app/server/trace/*.trc	Detailed trace files of execution of every client

../data directory

The `cyborghome/cyborgxx/data` directory contains the following files:

Filename	Description
taxfile	The Tax Authority file (U.S.).
taxfilec	The Tax Authority file (Canada).
taxfilee	The Tax Authority file for employer/employee taxes
cbsv	Contains source code for cbsvb, cbsvbt,cbsvo, and cbsvot
cbsv.ovr	Overrides to cbsv source.
cybmst	Contains source code for all the batch programs (P2EDIT, P4CALC, P5PRNT, P9CNVT, and O4CALC), COBOL and assembler subroutines, and report generators.
cybqtrus	U. S. Quarterly Processor UI programs
demo0105	The Solution Series System Control Repository in sequential format. This file contains option lists, tables, documentation, test data, and all Cyborg Scripting Language programs.
eprddio05	Special MAINTI05 file for including Enhanced Payroll Processing and DDI CheckList and Menu records via an additional installation process.
export.10	Output of jexport.bat
f1rstr	Source code for restore of FILE01
f2rstr	Source code for restore of FILE02
file4mnt	Upgrade script
file5mnt	Upgrade script
p05rdrqt.dat	File used for pulling quarterly rg's and qtrmisc.

Filename	Description
p5prnt.ovr	Overrides to P5PRNT source.
p5qtr.ovr	Overrides to P5QTR source.
p7comp.ovr	Overrides to P7COMP source
p9cbsvc.04	Report generator extract parameters for jsp9cbsvc
p9cnvt.ovr	Overrides to P9CNVT source.
p9cbsv.04	Report generator extract parameters for jxpcbsv.
p9cbsv.04	Report generator extract parameters for jxp9cbsv (Canada)
p9strt.04	Report generator extract parameters for jxp9strt.
rdbpgm	Relational only. Contains source code for rdbpgm0.cob, rdbpgm2.cob, rdbpgm3.com, and rdbpgm4.cob
re5204	--
re5205	--
re52m04	--
re52m05	--
vers80.ovr	System override file
BESS	Command Line Script called by the RSPAWNESS script to run the Enhanced Interactive Workforce System.
BHLD	Called by the RSPAWN script to call the bulk load program to copy extracted data into the tables specified.
BHRD	Called by the RSPAWN script.This script runs report and creates intermediate files.
BISLOCK	Checks to see if specified file is available for access.
BISW	Called by theRSPAWN script to run the Enhanced Is/Was report and create intermediate files.
BIWE	Called by the RSPAWN script to run the Enhanced Interactive Workforce Is/Was report, create intermediate files, then rename files in the users sub-directory.
BMRG	Runs an online batch payroll merge.
BPAY	Runs an online batch payrun.
BRPT	Runs an online batch report.
BXTR	Runs an online batch pay extract.
JPRT	--
JQRY	--
JRPT	--
P274	--
p247LAUNCH	One of the three scripts used to complete a pay process using 24/7 access

Filename	Description
p247PAY	One of the three scripts used to complete a pay process using 24/7 access
RESS	Required for Enhanced Reporting. Starts RSPAWNESS to revert control back to CBSVO.
RLCH	Required for Enhanced Reporting. Starts RSPAWN to revert control back to CBSVO.
RMRG	Required for Enhanced Reporting. Starts RSPAWMRG to revert control back to CBSVO.
RPAY	Required for Pay Process. Starts RSPAWNPAY to revert control back to CBSVO.
RSPAWN	Used for generating CBSVB output message logs on RLCH.
RSPAWNESS	Used for generating CBSVB output message logs on RESS.
RSPAWNMRG	Used for generating CBSVB output message logs on RMRG.
RSPAWNPAY	Used for generating CBSVB output message logs on RPAY.
RSXLAUNCH	One of three scripts that updates Reporting Administration labor and history tables in the Workforce Data Mart
RSXU	One of three scripts that updates Reporting Administration labor and history tables in the Workforce Data Mart
RSXUPD	One of three scripts that updates Reporting Administration labor and history tables in the Workforce Data Mart
u247	One of three scripts that perform a selective pay merge for 24/7 processing
u247LAUNCH	One of three scripts that perform a selective pay merge for 24/7 processing
u247UPD	One of three scripts that perform a selective pay merge for 24/7 processing

../prog directory

The /cyborghome/cyborgxx/prog directory contains the following files:

Filename	Description
cbsvb.cob	Non-relational program source code used to process The Solution Series in batch

Filename	Description
cbsvb.mf2	Non-relational program source code used to process The Solution Series in batch
cbsvb.pco	--
cbsvbr	Relational program source code used to process The Solution Series in batch
cybstdio.c	--
cbsvbr.pco	--
cbsvbt.cob	Non-relational trace program source code used to process The Solution Series in batch
cbsvo.cob	Non-relational program source code used to process The Solution Series online
cbsvot.cob	Non-relational trace program source code used to process The Solution Series online
CBSVRFT.CBL	Cobol script
cybgetkey.o	--
cybio.cob	Program for System Control Repository (FILE01) IO
cybsha1.o	--
p10sort.cob	Program that sorts data records in ascending order. Files used: p05in and p05out
p20cnvt.cbl	Cobol script
p45sort.cob	Program that sorts data records in ascending order. Files used: p40in and p40out
p80copy.cob	Program that adds carriage returns and line feeds to each record in a data file
p80sort.cob	Program that sorts data records in ascending order.
pfssort.cob	Program that sorts data records in ascending order Used in `f-segm`
p9cnvt.cob	Program source code used to extract any member from the cybmst file
rdbpgm0.cob	Creates databases, tables, and indexes to support the relational version of The Solution Series
rdbpgm2.cob	Creates databases, tables, and indexes to support the relational version of The Solution Series
rdbpgm3.cob	Creates databases, tables, and indexes to support the relational version of The Solution Series
rdbpgm4.cob	Creates databases, tables, and indexes to support the relational version of The Solution Series
repsort.cob	Cobol file - programs source code
sha.cob	Cobol file - programs source code

../runs directory

The \runs directory contains the following files:

File Name	Description
jbackem	Creates a sequential version of FILE1
jbldaky	Builds or rebuilds the Employee Name Alternate Key
jcf1rstr	Batch program to restore FILE01
jcf2rstr	Batch program to restore FILE02
jclean01	Removes extraneous information from the MAINTO Standard clean script
jclean3x	Reads a MAINTO file and removes any obsolete records
jcmpcbio	Compiles the delivered cybio program
jcmpcvbn	Compiles the delivered non-relational batch programs
jcmpcvbr	--
jcmpcvn	Compiles the non-relational batch programs as extracted from the CBSV file
jcmpcvr	--
jcmpp20	--
jcmprdb0	--
jcmprdb1	--
jcmpp9cv	Compiles P9CNVT program
jcmprft	--
jcmpsort	Compiles P10SORT.CBL, P25SORT.CBL, P80SORT.CBL, P80COPY.CBL and PFSSORT.CBL
jcmpsubr	--
jconv02	--
jconvert	Extracts, compiles, and links the CONVERT program
jconvna	Converts name and address data to new field format
jconvp20	Uses the FIXP20, CRUTDS, and CONVERT programs to create a 'clean' data dictionary
jcrossx	--
jcrtcyb	--
jcrtp20	--
jcrtp20c	--
jcrtpgms	--

File Name	Description
jcybio	--
jdemo01	Creates indexed System Control Repository from sequential FILE05 (DEMO0105)
jdemo01r	--
jdemo023	Reloads all CSL programs during the upgrade process
jdocxtr	--
jdscr08	--
jessxpt	Extracts employee data for Interactive Workforce
jexport	Exports the 'F1' and 'FTM' records from the System Control Repository; output FILE10 is used as input FILEIN2 in JCRTPGMS
jf-xref	Builds Field Name Table (F-NAME) cross references (RFT records)
jf1rstr	--
jf2rstr	--
jfultcl	--
jhrdemo	Extracts test data from the System Control Repository and populates fields in the online Employee Database
jhrdemoc	(Canada) Extracts test data from the System Control Repository and populates fields in the online Employee Database
jiswase	Produces the Interactive Workforce audit trail
jiswasx	Produces The Solution Series audit trail
jloadessgen	Loads the Interactive Workforce report generators
jloadgen	Builds a P20IN file that contains only report generators for the 24/7 access process
jloadhr	--
jlogexp	--
jlogext	--
jmainti	Updates the System Control Repository
jmainto	Compares current System Control Repository with original FILE05 (DEMO0105) and produces FILE10 (MAINTO10), which contains the differences found
jmakecl	Extracts option list values, field definitions, form security, and PC menu records from the System Control Repository
jmntf08	--

File Name	Description
jmntrun	Updates Labor and History records following the Batch pay calculation
jmtoanl	--
jmtocln	--
jmtoclr	--
jmtolod	--
jp20cnvt	--
jp20strc	Extracts Canadian report generators
jp20strt	Creates the P20IN Batch Master File for the first time
jpaymrg	Creates or updates the online Employee Database
jpaymrg_full	--
jpayrun	Calculates pay and produces checks, reports, and a combined register
jpayxtr	Pulls Time entries and adjustments from the online Employee Database and creates FILE12 (P20 Master), which will be the new P20IN file
jpfssort	Sample script to run segment layout report
jpopf01	RELATIONAL ONLY. Populates the RDBMS tables Cx through Xx
jprdemo	Extracts test data from the System Control Repository (Control File; FILE01) and populates fields in the online Employee Database; script is used when first building the employee database during installation
jprdemoc	Loads Canadian test companies
jpul_rdb	RELATIONAL ONLY. Extracts the CASE tool, RDBPGM0, RDBPGM2, RDBPGM3, and RDBPGM4
jpulcvn	Extracts non-relational batch programs CBSVB, CBSVBT, CBSVO, and CBSVOT from the CBSV file
jpulcvr	Extracts relational batch programs CBSVB, CBSVBT, CBSVO, and CBSVOT from the CBSV file
jpulcvs	Extracts the CVSB COBOL programs
jqtrrun	Runs the Quarterly Processor
jrebuild	Recreates System Control Repository using the output from JBACKEM.BAT

File Name	Description
jreload	Reloads (Solution Series compile routine) Cyborg Scripting Language programs in the System Control Repository
jreport	Extracts reports from System Control Repository and the Employee Database
JRPTMNU	Build a menu containing the delivered report groups for Enhanced Payroll Reporting
jupdateU	Upgrade script
jupdcybm	Updates the CYBMST file with program updates or report generators
jupdgen	Upgrade script
jxcybmst	Extracts and compiles CYBMST programs (O4CALC, P2EDIT, P4CALC, P5PRNT, and P9CNVT)
jxessrptgen	Extracts the Interactive Workforce report generators
jxo4calr	Compile relational O4CALC
jxp5qtr	Extracts P5QTR from CYBMST, compiles and links machine specific subroutines
jxp7comp	Extracts and compiles P7COMP from CYBMST
jxrept20	Extracts report generator 20 from CYBMST in order to add new Organization Control Number values (companies) to the P20IN Batch Master File
jxrptgen	Extracts report generators from CYBMST
jxrptqtr	Extracts Quarterly Processor report generators
online	Initiates the online Solution Series system by executing the CBSVO program
onlineE	--
onlinet	Initiates the trace online Solution Series system by executing the CBSVOT program
rj	Executable that automatically launches a program and displays the log output immediately after the program completes its run

A P P E N D I X   C

Creating Separate Environments on the Server for the Client

In This Appendix

Phase 1:Performing server configuration

Configuration Chart

Complete the following tasks depending upon the type of environment you are configuring:

If *existing* configuration is …	And *new* configuration is …	Perform
non-relational	non-relational	Create a unique directory under `cyborghome`. Copy programs from the first installed environment. Add additional environments to the Cyborg environment configuration file. Grant read, write, execute permissions to user ID.. 'CYBORG' for each new subdirectory (runs, work, list, prog, data)
relational	non-relational	Create a unique directory under `cyborghome`. Execute script jbackem in relational environment for a sequential backup of FILE01. Execute script payxtr.bat. All tasks in Chapter 2, incorporating your overrides from the relational environment. For task, Create System Control Repository, use the output of jbackem (backem.10) as input to script jrebuild. Execute paymrg.bat with the p20.xtr created from payxtr. Add additional environments to the environment configuration file. Grant read, write, execute permissions to user ID 'CYBORG' for each new subdirectory (runs, work, list, prog, data).

If *existing* configuration is …	And *new* configuration is …	Perform
non-relational	relational	Create a unique directory under cyborghome.
		Execute script jbackem in non-relational environment for a sequential backup of FILE01.
		Execute script payxtr.bat.
		Execute script jexport in non-relational environment to extract F1 and RFM records.
		All tasks in Chapter 3, incorporating your overrides from the non-relational environment. For task, Execute the Case Tool, use the output of jexport (export.10) as input to script jcrtpgms. For task, Create System Control Repository, use the output of jbackem (backem.10) as input to script jrebuild.
		Execute paymrg.bat with the p20.xtr created from payxtr.
		Add additional environments to the environment configuration file.
		Grant read, write, execute permissions to user ID 'CYBORG' for each new subdirectory (runs, work, list, prog, data).

Phase 2: Performing client configuration

Configure a new environment connection at the client

To configure a new environment connection at the client, perform the following steps:

1.   Access the Connection Editor dialog box

Access this dialog box by selecting:

Start ▶ Programs ▶ Hewitt Associates ▶ The Solution Series xxx ▶ Connection Editor

2.   Click New

The New Connection entry dialog box displays.

3.   Enter a Connection Name

Identify a title for the connection you will configure between the client and the server. For example, if your production environment is called CYBPROD and your test environment is called CYBTEST, you can assign a more user friendly name here such as Solution Series 5.x.x Production System. If a user requires multi-environment access, additional environments can be set up later. This label must not contain characters '\\/:*?\"<>_'.
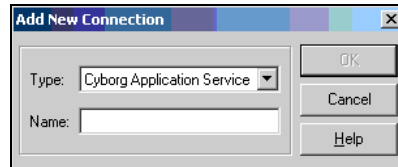
Note   You must create a folder on the server which will match the Connection Name that you enter here. See Build the Client Data File (next section). Create a connection subdirectory at the client.

4.   Click OK

The Connection Editor dialog box displays a new connection with default settings, ready for configuration.

5.   Type the configuration details

Type the following configuration details under Connection Properties on the Connection Editor dialog box:

Item	Description
Host	Identify the system name of the UNIX server, as identified on the network.

Item	Description
Port	Identify the port address of the UNIX server. Port address 9888 has been registered for the Cyborg Application Server (CAS). Note You must also identify the 9888 port address on the server by editing the /etc/services file, as described in Chapters 3 and 4, Phase 6: Install and configure Cyborg Application Server (CAS) daemon.
Environment	Identify the environment name (up to 8 characters). Examples are: CYB52PROD, CYB52TEST. Note This same environment name must also be entered in the cybenv.cfg file on the server, as described in Appendix E: Administering the Cyborg Application Server (CAS) Daemon.
Be Cyborg User	Accept the default entry of Yes to enable the Cyborg User.
Encryption	The default entry is No. If Yes is entered, the sign-on data sent between the client and the server will be encrypted.
ST Application ID	The application name and ID number for The Solution Series application. This application ID is configured to 2 as identified in cybapp.cfg on the server.
FILE01 Application ID	The application name and ID number for the CYBIO application. This application ID is configured to 3 as identified in cybapp.cfg on the server.
Server Username	Leave this field blank.
Server Password	Leave this field blank.

6. **Click Exit**

The new connection has been configured between the server and the client.

Build the Client Data File

This task details how to build the Client Data File using the Build FileCL utility.

Note   *You can build one Client Data File at the server for each environment, then copy it to each client. The Client Data File can also be built by deleting the existing FILECL32. When a person logs on to the system after this has been deleted, The Solution Series will automatically build a new Client Data File.*

Run Export Client File utility (makecl) on the server
Script used: jmakecl

To obtain the source file for the Client Data File, execute the jmakecl script from the $runs subdirectory. For example:

jmakecl

Review the log to determine if there were any errors.
The makec110 file (FILE10) must then be copied to a local client PC.

> *Note*   *Run this script in each environment for which you want to create a client data file.*

Test the connection (on the client)

1.   Launch The Solution Series
Select:
Start ▶ **P**rograms ▶ The Solution Series ▶ The Solution Series

The Welcome screen displays.

2.   Select the Start button on the Welcome screen
The sign on dialog box displays.

3.   Sign on as the Security Officer
Select the environment you want to access, enter your user name and password.

4.   Click OK
The work area for The Solution Series displays.

5.   Sign off The Solution Series

A P P E N D I X   D

ORACLE Database Considerations

In This Appendix

Overview

This section has been provided to assist customer DBAs to properly create the database and minimize any rework in the future or troubleshoot database problems.

Understanding the Solution Series Database

There are several things you should understand when working with the database. This section explains the specifics of the database.

Tablespaces

Following are the database tablespaces created to hold data and index objects:

Tablespace 0	Temporary tablespace the Cyborg user will use
Tablespace 1	Company data + Labor/History Indexes
Tablespace 2	Other Record data + Employee Indexes
Tablespace 3	Tax data + Tax Indexes
Tablespace 4	Employee data + Other Record Indexes
Tablespace 5	Labor/History data + Company Indexes
Tablespace 6	Option List/Application Tables
Tablespace 7	Option List/Application Table Indexes
Tablespace 8	Z-TABLE
Tablespace 9	Z_TABLE Indexes
Tablespace A	TABLE01 + ZZ2IND
Tablespace B	TABLE01 + ZZ2IND Indexes

User account

An ORACLE user account should be created to own the Solution Series database objects. Use the following guidelines:

- Minimum database privileges this user should be granted.
- Create database tables, views, synonyms, roles, procedures, and triggers.
- Select, insert, update, and delete for database tables and views.

Enough memory on machine

Before tuning memory for ORACLE, ensure that enough memory resides on your machine for the following:

- Operating system
- Assortment of support mechanisms for the operating system
- Database engine, tools, and shadow processes for your version of ORACLE
- Software that coexists with ORACLE on your machine
- Network of operating system buffers
- ORACLE databases that coexist on your machine
- Memory overhead per user on the system
- Operating system overhead for supporting the read and write requests of all of those users

Tuning ORACLE with INIT.ORA parameters

To see the current value of your INIT.ORA parameters, you can no longer rely on reading the INIT.ORA file; you should select all of the parameter names and values from V$PARAMETER instead. The setting of the INIT.ORA Parameters customizes the performance of each ORACLE instance to its particular needs. While default settings often yield adequate performance, the peak performance ORACLE is capable of delivering can often be attained only by careful tuning of each parameter.

While most of the parameters can be adjusted only after the database is in use, the following parameters can be adjusted immediately upon installation:

- SYSTEM GLOBAL AREA (SGA)
    - DB_BLOCK_SIZE
    - DB_BLOCK_BUFFERS
    - SHARED_POOL_SIZE
- OPEN_CURSORS
- PROCESSES
- SORT_AREA_SIZE
- DML_LOCKS
- LOG_BUFFER
- ROLLBACK_SEGMENTS

The parameters are listed in order of maximum performance gain.

SYSTEM GLOBAL AREA (SGA)

It is important that the SHARED_POOL_SIZE and DB_BLOCK_BUFFERS account for 90% of the SGA total size. In addition, the SGA should never take over 50% of the available memory. In a SVRMGR session, enter the following to determine the SGA values:

sho SGA

DB_BLOCK_SIZE

The size of a database block in bytes. We suggest using the default value of 2048.

DB_BLOCK_BUFFERS

The number of database blocks cached in memory. Each buffer in the cache contains one ORACLE block. The larger the cache, the more data ORACLE can hold in memory. If the data is not in memory, ORACLE issues the needed I/O request to obtain the data, which is the slowest operation a computer can perform. Set this value to the maximum number of buffers that could be added without causing paging.

SHARED_POOL_SIZE

The size in bytes of shared pool. If the ratio of reloads to pins exceeds 1 percent, you should increase this parameter. This can be determined by a simple query:

SQL>SELECT (SUM(reloads/SUM(pins)) * '100 Miss' % from V$LIBRARYCACHE;

OPEN_CURSORS

This parameter is the maximum number of cursors that a user can have open at one time. To fully use the higher value for SHARED_POOL_SIZE, you may also want to increase the number of cursors available to each user (OPEN_CURSORS).

PROCESSES

This parameter limits the number of users who can concurrently access the instance. This parameter does not affect performance but is a useful starting point in defining expected requirements for ORACLE. Keep in mind that the background processes are included in this number and if the application spawns processes recursively, all these spawned processes count.

SORT_AREA_SIZE

This is the amount of memory per user process that is allocated for sorting. Size your SORT_AREA_SIZE to fit the need of the users. This is a big user of memory and also a big help with performance.

DML_LOCKS

This parameter is the maximum number of locks that can be placed on all tables by all users at one time. Experience has shown this parameter should be set high, as this parameter has no effect on performance.

LOG_BUFFER

This parameter is the number of bytes that are allocated to the redo log buffer in the SGA. If the ORACLE system is processing many in-process transactions, this parameter should be increased to reduce I/O to the redo logs.

ROLLBACK_SEGMENTS

This parameter is a list of all the rollback segments available to user processes. The system rollback segment should never appear in this parameter's list. All of the user rollback segments should be the same size since they are allocated randomly. Rollback segments should be large enough to contain all of the rollback information for any anticipated transaction. Always name your rollback segments in the initialization parameter file. Always place your rollback segments in their own tablespace.

☞ *Refer to 'Managing Rollback Segments', later in this appendix.*

Space Management

Space is needed for the following objects to extend tables and indexes, rollback segments, and temporary tables:

Tables and indexes

This is caused by the said objects needing additional space to satisfy an insert or update.

Rollback segments

If the culprit is a rollback segment, the error ora-1562 'failed to extend rollback segment (id = %s)' will always precede the ora-1547. The ora-1562 is telling us that it could not extend the rollback segment, and the reason is the ora-1547—not enough space.

Temporary tables

These are tables created by the ORACLE kernel to do a sort on behalf of the user. A user can tell that he is running out of space for a temporary table, based on the operation he/she is performing (such as creating an index, doing a query with an order by, or a lengthy join statement). The temporary tablespace the user will use can be seen by performing the following query:

SQL>select temporary_tablespace from sys.dba_users where username='<USERNAME>';

If the space being used seems too large, you may want to investigate the default storage for the temp tablespace—it is possible that the defaults are too small. To see the default storage, perform the following query:

SQL>select inital_extent, next_extent, min_extents, pct_increase fromsys.dba_tablespaces

 where tablespace_name='<NAME>';

Adjustments can be made to the default storage of the tablespace by issuing the following command:

SQL>alter tablespace <NAME> storage (inital xxx next xxx....);

Space can be added to a tablespace using the 'ALTER TABLESPACE' command (full syntax below). This statement will create a database file on disk and enlarge the existing tablespace. The statement can be performed on all tablespaces (including system) without shutting down the database or taking the tablespace offline. Immediately following the completion of the statement, the space is available.

SQL>alter tablespace <TABLESPACE_NAME> add datafile '<PATH/FILENAME>' size <size_of_file> reuse;

To get an idea of the naming conventions or locations for existing files, perform the following query:

```
SQL>select file_name from sys.dba_data_files where
tablespace_name='<NAME>';
```

Understanding and resolving common Oracle sizing errors with tablespaces

Error 01658: Unable to create INITIAL extent for segment in tablespace %s

Cause:	Failed to find sufficient contiguous space to allocate INITIAL extent for segment being created.
Action:	Use ALTER TABLESPACE ADD DATAFILE to add additional space to the tablespace or retry with a smaller value for INITIAL.

ORACLE will ALWAYS try to allocate CONTIGUOUS space. Although the tablespace may have enough free space, if it is not contiguous, the error will occur. To see if you have enough contiguous space in the tablespace, perform the following query:

SQL>select max(blocks) from sys.dba_free_space where tablespace_name='<NAME>';

This will return one record that shows the biggest chunk of space free in the tablespace in question. This number will be lower than the one returned by the error. If you wish to compare the contiguous space with total space, perform the following query:

SQL>select sum(blocks) from sys.dba_free_space where tablespace_name='<NAME>';

This also returns one record. This value can be compared to the record above to see how much of the total space is contiguous.

Understanding and resolving common Oracle sizing errors with tables

Error 01631: Max # extents (%s) reached in table %s.%s

Cause:	A table tried to extend past maxextents.
Action:	Recreate the table with larger initial, next, and pctincrease extents. If this is not possible, check if seIf maxextents storage parameter is less than system allowable max, then raise this value and consider upping the pctincrease value as well.

Error 01653: Unable to extend table %s.%s by %s in tablespace %s

Cause:	Failed to allocate an extent for table segment in tablespace
Action:	Use the ALTER TABLESPACE ADD DATAFILE statement to add one or more files to the tablespace indicated.

ORACLE uses the logical 'tablespace' unit. However, the physical aspect of the tablespace unit is the datafile. The datafile, which is created physically on disk, is where all objects within that tablespace reside. In order to add space to the tablespace, you must add a datafile.

Understanding and Resolving common Oracle sizing errors with indexes

Error 01632: Max # extents (%s) reached in index %s.%s

Cause:	An index tried to extend past maxextents.
Action:	Recreate the index with larger initial, next, and pctincrease extents. If this is not possible, check if seIf maxextents storage parameter is less than system allowable max, then raise this value and consider upping the pctincrease value as well.

Error 01654: Unable to extend index %s.%s by %s in tablespace %s

Cause:	Failed to allocate an extent for index segment in tablespace.
Action:	Use the ALTER TABLESPACE ADD DATAFILE statement to add one or more files to the tablespace indicated.

ORACLE uses the logical 'tablespace' unit. However, the physical aspect of the tablespace unit is the datafile. The datafile, which is created physically on disk, is where all objects within that tablespace reside. In order to add space to the tablespace, you must add a datafile.

Understanding and resolving common Oracle sizing errors with rollback segments

Error 01628: Max # extents (%s) reached for rollback segment %s

Cause:	Tried to extend rollback segment already at maxextents value.
Action:	Recreate the rollback segment with larger initial, next, and pctincrease extents. If this is not possible, check if self maxextents storage parameter is less than system allowable max, then raise this value and consider upping the pctincrease value as well.

Error 01650: Unable to extend rollback segment %s by %s in tablespace %s

Cause:	Failed to allocate an extent for rollback segment in tablespace.
Action:	Use the ALTER TABLESPACE ADD DATAFILE statement to add one or more files to the tablespace indicated.

ORACLE uses the logical 'tablespace' unit. However, the physical aspect of the tablespace unit is the datafile. The datafile, which is created physically on disk, is where all objects within that tablespace reside. In order to add space to the tablespace, you must add a datafile.

Managing rollback segments

A rollback segment consists of contiguous multi-block pieces called extents. The segment uses these extents in an ordered circular fashion, moving from one to the next after the current extent is full. A transaction writes a record to the current location in the rollback segment and advances the current pointer by the size of the record.

To determine a general rollback segment configuration, balance the number of segments against the size of each segment, such that the space needed will fit into the available disk space.

Make sure that there are enough segments to avoid contention as processes access them. But also make sure that individual rollback segments are large enough for their transaction load.

Balancing transaction requirements

The next section discusses balancing these two requirements.

- A transaction can only use one rollback segment to store all of its rollback (undo) records.
- Multiple transactions can write to the same extent.

There are two issues that need to be considered when deciding if your segment is large enough.

First, make sure that transactions will not cause the head to wrap around too fast. This causes the segment to extend in size, per the principles mentioned above.

Second, if you have long running queries that access data that frequently changes, make sure that the rollback segment does not wrap around to prevent the construction of a read consistent view.

☞ *Refer to the ORACLE Database Administrator's Guide for discussions on 'read consistency' on 'avoiding the snapshot too old error'.*

The size needed for a rollback segment depends directly on the transaction activity of your database. Be concerned about the activity during normal processing of the database, not with rare or semi-frequent large transactions. These special cases are to be dealt with separately.

The number of rollback segments needed to prevent contention between processes can be determined with the help of the monitor rollback display and the use of the V$WAITSTAT table.

Undo headers may occur if there are not enough rollback segments to support the number of concurrent transactions. The following V$WAITSTAT query will display the number of waits since instance startup:

SQL>SELECT * FROM V$WAITSTAT WHERE CLASS = 'undo header';

To find out the size and number of rollback segments needed for normal processing on the database, you need to do some testing. A good test is to start with small rollback segments.

Allow your application to force them to extend. Here are the steps to run such a test:

1. Create a rollback segment tablespace.
2. Create a number of rollback segments in the tablespace.
3. Create the rollback segments so that all extents are the same size. Choose an extent size that you suspect will need between 10 to 30 extents when the segments grow to full size.
4. Each rollback segment should start with two extents before the test is run. This is the minimum number of extents any rollback segment can have.
5. Activate only the rollback segments that you are testing by making the status 'online'. The only other segment that should be 'online' is the system rollback segment.
6. Run transactions with a load typical of the application.
7. Watch for rollback segment contention.
8. Watch for the maximum size a rollback extends to.

The maximum size any one of the rollback segments reaches during the test is the size you want to use when configuring. This size we will call the 'minimum coverage size'. If you see contention, adjust the number of segments and rerun the test. Also, if the largest size requires fewer than 10 extents, or more than 30, it is a good idea to lower or raise the extent size respectively, and rerun the test.

For sizing rollback segment extents, we strongly recommend that each extent be of the same size. In fact, we also suggest that the size of the rollback tablespace is some multiple of the common extent size. The number of extents for an individual segment should be around 20.

In the rollback segment storage clause, please use the OPTIMAL parameter. OPTIMAL sets an optimal size in bytes for a rollback segment. It can be specified in kilobytes or megabytes. ORACLE will dynamically deallocate extents in the rollback segment to maintain the optimal size.

NULL means that ORACLE never deallocates the rollback segment extents, and this is the default behavior. You must supply a size greater than, or equal to, the initial space allocated for the rollback segment by the MINEXTENTS, INITIAL, NEXT, and PCTINCREASE parameters.

Extent deallocation is expensive in regards to performance. This means that an OPTIMAL setting may decrease performance if it is too low.

Changing an ORACLE user's password

You can use the ALTER USER command as a DBA or as the user itself to accomplish this task.

SQL>CONNECT userid/password;

where *userid* is your database userid or the userid of the DBA and *password* is your current password or the password of the DBA.

SQL>ALTER USER john IDENTIFIED BY test;

Dropping the Database

Dropping a database is not supported by ORACLE. However, taking the tablespaces offline, dropping each tablespace, dropping the Cyborg user, and deleting all related data, initialization, and control files will accomplish this. For each of the 8 tablespaces created for the database, perform the following two commands:

SQL>alter tablespace <TABLESPACE_NAME> offline;
SQL>drop tablespace <TABLESPACE_NAME> including contents cascade constraints;

Then drop the Cyborg user:

SQL>drop user <USER> cascade;

Delete all related data, initialization, and control files in the operating system.

A P P E N D I X   E

Administering the Cyborg Application Server (CAS) Daemon

In This Appendix

Overview of the Cyborg Application Server Daemon

This appendix introduces you to the Cyborg Application Server (CAS) daemon and includes information on how to:

- Start CAS
- Configure a new environment on the server
- Use the CAS Manager to perform administrative functions
- Monitor CAS using the system log File
- Troubleshoot any problems you may encounter when installing CAS
- Enable and disable tracing
- Stop CAS (terminate the CAS daemon entirely by forcing CAS to exit)
- Uninstall CAS

What is the Cyborg Application Server (CAS) daemon?

Overview of CAS

The Cyborg Application Server (CAS) is a server that provides network transparency of clients for server applications. It allows an existing application, such as CBSV, to become the server in a client/server architecture with almost no modification to that application.

The client connects to CAS across the network via TCP/IP. The client, such as The Administrative Client, is fully aware of CAS and sends requests and receives responses via messages. Among other things, these messages ask CAS to start the server application, send input to the application, receive data from the application, and receive responses from the server.

CAS supports multiple server applications on a single system. Many clients can be connected to an environment via CAS at the same time. Furthermore, CAS supports multiple environments on a single server. Each client can run any number of server applications available to it.

'listening' CAS and 'talking' CAS

The CAS works by creating copies of itself to allow multiple clients to access the server applications.

The CAS daemon is informally known as 'listening' CAS because it listens for incoming client requests from the network. When it receives a request from the client, it creates a child process (informally known as 'talking' CAS) to interact with the client application:

- The CAS daemon begins its execution as 'listening' CAS by waiting for a client connection from the network.

- When 'listening' CAS detects that a client has connected, it immediately creates a copy of itself; this copy is 'talking' CAS.

- After creating the copy, 'listening' CAS is free to listen for new clients, while 'talking' CAS will be responsible for starting the server application and handling all subsequent communication with that client, including passing I/O between the client and server application.

- At any time, there is only one 'listening' CAS on the UNIX system, while there can be any number (including zero) 'talking' CAS processes.

Major components of CAS

The following diagram gives you an overview of the major components of the Cyborg Application Server as it is implemented on a UNIX platform. It shows the processes that interact with UNIX CAS and their I/O relationships.
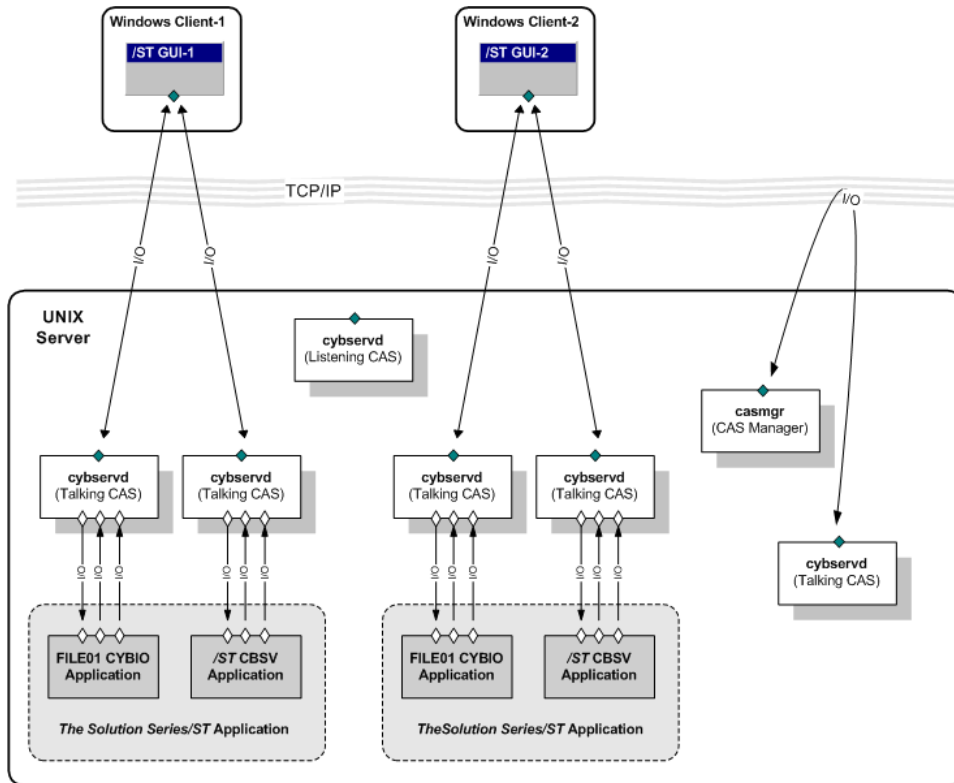
Explanation of the cas script

To launch the Cyborg Application Server (CAS) daemon, you need to execute the CAS script. The following is the delivered CAS script:

```
#!/bin/ksh
#
# {cyborg home}/app/server/cas script 1.16 for UNIX CAS 1.04
```

```
#
# The Cyborg Application Server files and directories all reside in the
# same directory as this script.
#
cd `dirname $0` || exit 1
casdir=`pwd`
daemon=$casdir/cybservd
appcfg=cybapp.cfg
envcfg=cybenv.cfg
tracedir=trace
maxusers=2000
#
# Enter MicroFocus (and Oracle) environment variable lines here.
#-------------------------------------------------------------
#
#-------------------------------------------------------------
#
#
# Check that the proper files and directories exist
#
assert()
{
    if command test ! $1
    then
         echo "The current directory is `pwd`"
         echo "Error:" $2
         exit 1
    fi
}
assert "-e $daemon" \
       "The CAS executable \"$daemon\" does not exist"
assert "-f $daemon" \
       "The CAS executable \"$daemon\" is not a regular file"
assert "-x $daemon" \par      "The CAS executable \"$daemon\" does not have
executable permission"
assert "-e $appcfg" \
       "The application configuration file \"$appcfg\" does not exist"
assert "-f $appcfg" \
       "The application configuration file \"$appcfg\" is not a regular file"
assert "-e $envcfg" \
       "The environment configuration file \"$envcfg\" does not exist"
assert "-f $envcfg" \
       "The environment configuration file \"$envcfg\" is not a regular file"
assert "-e $tracedir" \
       "The trace file directory \"$tracedir\" does not exist"
assert "-d $tracedir" \
       "The trace file directory \"$tracedir\" is not a directory"
#
# Calculate the number of connections required.
# - Each user requires two sessions:  one for online and one for cybio
# - The administrator should also get a session
#
peruser=2
maxconn=`expr $peruser \* $maxusers + 1`
#
# By default, CAS creates the system log and trace files with read/write
# privileges for everyone (user, group, other).  Set the permission mask
# so that trace files have the desired permissions (e.g., 066 to only allow
# access by the file owner or 022 to allow anyone to read the file, but
# only write by the owner).
```

```
#
umask 066
#
# Launch the daemon
#
$daemon -a$appcfg -e$envcfg -T$tracedir -c$maxconn
```

The `cas` script:

- Is delivered in the `/cyborghome/app/server` directory.
- Contains the cybservd command line that launches the CAS daemon.
- Specifies the location of the cybenv.cfg, cybapp.cfg, system log and trace files. As delivered, the default location of these files is: `/cyborghome/app/server` directory.

Note: *The Micro Focus and ORACLE environmental variables that were identified in Chapters 3 and 4 (Task 4) must be included where shown in this script before CAS is started.*

CAS script port
To edit the CAS script to include reference to new port

$daemon -p*nnnn* -a$appcfg -e$envcfg -T$tracedir -c$maxconn

where *nnnn* is the new port number.

Explanation of cybapp.cfg

The following is the delivered cybapp.cfg data file. It should not be changed.

```
#
# cybapp.cfg:  application configurations for the Solution Series/ST
#
# Each definition must fit on a line, and the fields in each definition
# are separated by colons.
#
# Name            Type                    Description
# ------------------------------------------------------------------
# App ID          Number: greater than 2  ID from the app configuration file
# Enabled         Boolean: 'Y' or 'N'     Whether the environment is enabled
# Name            String: up to 32 chars  Name of the application
#
2:Y:/ST Application
3:Y:FILE01 Application
```

Configuring a new environment on the server

You need to configure an environment for Solution Series users by editing the cybenv.cfg file. The cybenv.cfg file defines the two environments necessary for each client application to run The Solution Series:

```
#
# cybenv.cfg:  environment configurations for the Solution Series/ST
#
# The string "$runs" on the last few lines must be replaced with the
# absolute path to the appropriate directory for that environment.
#
#
```

```
#
# Each definition must fit on a line, and the fields in each definition
# are separated by colons.
#
# Name          Type                    Description
# ----------------------------------------------------------------
# App ID        Number: greater than 2  ID from the app configuration file
# Environment   String: 1-8 characters  Environment where the app is used
# Enabled       Boolean: 'Y' or 'N'     Whether the environment is enabled
# Idle timeout  Positive number or -1   Timeout in minutes before application
#                                       is shut down by the server; use a
#                                       value of -1 to disable the timeout.
# Working dir   String: directory       Starting directory of host app
# Executable    String: filename        Program to launch as the host app
# Parameters    String (optional)       Parameters to the host app
#
2:env1:Y:-1:$runs:$runs/online
3:env1:Y:-1:$runs:$runs/jcybio
```

Each environment definition (highlighted in the script above) has six fields, with the fields separated by a colon (:). All fields must appear in their defined order. The following diagram identifies those six fields:
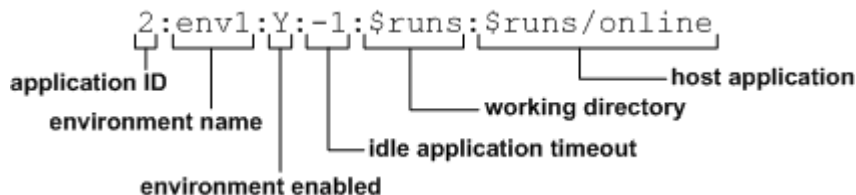
The following table describes the environment definition fields and how to edit the cybenv.cfg file:

Field	Description	How to edit cybenv.cfg
application ID	This field is the application ID of the application with which this environment is associated. This ID must correspond to the application ID in the application configuration file (cybapp.cfg).	Enter: 2 for /ST application (online) 3 for FILE01 application (jcybio) IMPORTANT: Each environment must have two entries in the cybenv.cfg file: One for an application ID of 2 (for the /ST application), and A second for an application ID of 3 (for the FILE01 application).

Field	Description	How to edit cybenv.cfg
environment name	This field identifies the environment name (up to 8 characters). Examples are: CYB52PROD, CYB52TEST. All environment names for each application ID must be unique and are not case-sensitive. For example, CYB52PROD and CYB52prod are not unique and are, in fact, the same environment.	To set up a Solution Series prod environment, enter: 2:prod:… 3:prod:… This environment will be the same environment as set in the configuration using the Connection Editor on the client workstations, as described in Chapter 5: Installing and Configuring The Solution Series on the Client.
environment enabled	This field is a denotes whether the application is enabled with values of Y or N. If the application is disabled, then any request to execute it will be refused.	Enter a Y to signify that each application is enabled. 2:prod:Y:… 3:prod:Y:…
idle application timeout	This field controls how long a program may remain open when there is no activity. The amount of time is specified in minutes. If there is no communication between the client and application program within this period of time, the program is terminated. A setting of -1 disables this feature.	The timeout value is either a positive number (1, 2, 3, and so forth) or is disabled with a value of -1. As delivered, the /ST application has a value of 20 in this field, indicating that it will be terminated after 20 minutes of inactivity. The FILE01 application is delivered with a value of –1 in this field, indicating that it will never be terminated due to inactivity.
working directory	This field refers to the directory path of the working directory. Each user should have executable permission to this directory.	Change the path to reflect the current installation by replacing $runs with the full path.
server application	This field refers to the script that will be launched by CAS for each application. Each user should have executable permission for this file.	Change the path to reflect the current installation by replacing $runs with the full path to each script that will be launched by CAS.

Using the CAS Manager

Overview of the CAS Manager

The CAS Manager is a standalone program that is used to manage any version of UNIX CAS from anywhere across the network. The CAS Manager is a command-line application that is driven by the options selected on the command line.

The CAS Manager allows an administrator to perform the following tasks:

- Determine whether CAS is running
- Determine the version of CAS and the operating system on which CAS is running
- Determine whether the CAS service is enabled (in other words, whether clients are allowed to start applications)
- Disable the CAS service to disallow future clients from starting applications
- Enable the CAS service to allow future clients to start applications
- Obtain a list of sessions
- Terminate a single server application, such as CBSVO/T or CYBIO, for a specific client
- Terminate all server applications
- Terminate the CAS service entirely by terminating the CAS daemon
- Tracing the execution of CAS as it handles a single server application for a given client
- Trace the execution of every CAS process
- Disable the tracing of CAS for a single server application for a given client
- Disable the tracing of every CAS process
- Trace the execution of CAS for subsequent clients
- Disable the tracing of CAS for subsequent clients

Passwords
Prompting for a password

If a switch requires a password and none is provided on the command line, then the user is prompted for a password (this is the password for the user ID 'cyborg'), similar to the `su(1)` or `rlogin(1)` commands. If a switch requires a password and one is provided on the command line, then the user is not prompted.

For example, the `-tracedefault` switch requires a password:

```
casmgr -tracedefault
Password: _
```

If the password prompt is used, the typed password does not echo on the screen and cannot be redirected from a file; it must be typed from the console.

Password security

We advise that you do *not* use the `-password` modifier unless CAS Manager is needed in a script. Instead, let CAS Manager prompt for a password, as shown above. This method negates the need to remember which commands require passwords.

CAS Manager syntax

The `casmgr` command invokes the CAS Manager. To use the CAS Manager, use the syntax shown below:

$ casmgr -*switch* -port:*nnnn* -password:*cyborg user's password*

Note   *If you want a description of the syntax or a brief description of each switch, run the CAS Manager without any switches specified.*

To find out/do this	Use this switch
Is CAS is running now?	-isrunning
Is CAS service is enabled now?	-isenabled
Disable CAS service	-disable
Enable CAS service	-enable
What version of CAS or CAS Manager is this?	-version
What clients are connected?	-sessions
Trace a single session on the talking CAS	-tracesession:yyy
Disable the trace on a single session on the talking CAS	-notracesession:yyy
Trace all CAS processes	-traceall
Disable the trace on all CAS processes	-notraceall
What is the default trace setting now for new CAS processes?	-istracedefault
Enable tracing for all new CAS processes	-tracedefault
Disable tracing for all new CAS processes	-notracedefault
Terminate a single host application for a single session	-killsession:yyy
Terminate all host applications	-killtalking

CAS switch details

The following describes the various switches that can be used with the `casmgr` command. The switches are in alphabetical order:

```
-disable

-enable

-isenabled

-isrunning

-istracedefault

-killdaemon

-killsession:yyy

-killtalking

-notraceall

-notracedefault

-notracesession:yyy

-sessions

-traceall

-tracedefault

-tracseession:yyy

-version
```

-disable
Disable CAS

Use this switch to disable the CAS daemon by disallowing future clients from starting applications:

casmgr **-disable**

This will place CAS in the disabled state. CAS will still be running, but will refuse any requests to start server applications. Previously connected clients will continue to run normally.

Note: *This command requires a password.*

-enable
Enable CAS

Use this switch to enable the CAS daemon, allowing future clients to start applications:

casmgr **-enable**

This will place CAS in the enabled state. CAS will accept requests to start server applications.

Note: *This command requires a password.*

-isenabled
Determine whether CAS is enabled or disabled

Use this switch to determine whether the CAS daemon is enabled (that is, whether clients are allowed to start applications):

casmgr **-isenabled**

This will print whether the CAS daemon is enabled or disabled. When the daemon is enabled, CAS will accept requests to start server application programs. When the daemon is disabled, CAS will still be running but will refuse any requests to start server applications. Use the -enable and -disable switches to enable and disable CAS.

-isrunning
Determine whether CAS is running

Use this switch to determine whether CAS is running:

casmgr -isrunning

This will determine whether there is a CAS process running on the server.

-killdaemon
Terminate the CAS service

Use the -killdaemon switch to terminate the CAS daemon entirely and force the CAS process to exit:

casmgr **-killdaemon**

The daemon cannot be restarted using the CAS Manager, as there is no longer any daemon to answer CAS Manager commands. CAS must be re-started by the system administrator, cron, or other external UNIX facility.

Use the 'determine whether CAS is running' switch (-isrunning) to verify that CAS has terminated.

Note: *This command requires a password.*

-killsession
Terminate a single server application program for a specific client

Use this switch to terminate a single server application program for a specific client:

casmgr **-killsession:sessionID**

This will terminate a single server application program for a specific GUI client. The application program is identified with a session ID, as determined by the 'obtain a list of connected sessions' switch (-sessions).

Use the 'obtain a list of connected sessions' switch (-sessions) to verify that the application program was terminated.

Note: *This command requires a password.*

Monitoring the CAS daemon
Tracing
CAS can create trace files to debug a CAS installation and monitor the messages that are sent and received between the client and server application.

- CAS turns on the trace mechanism in response to an Enable Tracing message.
- CAS turns off the trace mechanism in response to a Disable Tracing message.

Tracing can be enabled or disabled by using various switches with the CAS Manager (casmgr) command, including:

```
-notracesession
```
```
-notraceall
```
```
-notracedefault
```
```
-tracesession
```
```
-traceall
```
```
-tracedefault
```

Using the system log File
What is the system log file?
The system log is a single, system-wide log shared by every CAS instance on a particular machine for recording significant events (such as newly connected clients, disconnected clients, and so forth) that occur during CAS execution.

Because it is important for an administrator to see the sequence of events that occur while CAS executes, all events are recorded in the system log file.

Filename

The filename of the system log file is `system.log` and it is found in the `/cyborghome/app/server` directory.

The following is an example of a system log file:

```
2004/12/21  08:53:29  cybservd:1153   Info   15456   Initialize service
(Service) (None)                /ST 5.2 for UNIX:  CAS version

1.00
2004/12/21  08:53:29  cybservd:1157   Info   15456   Initialize service
(Service) (None)                Compiled on HP-UX version A,

release B.10.20
2004/12/21  08:53:29  cybservd:1169   Info   15456   Initialize service
(Service) (None)                Executing on machine bldrux1:

HP-UX version A, release

B.10.20
2004/12/21  08:53:29  sharmem:353    Info   15458   Initialize service
(Service) (None)                Allocated semaphore (semaphore

ID=299)
2004/12/21  08:53:29  sharmem:357    Info   15458   Initialize service
(Service) (None)                Allocated shared memory (shared

memory ID=4208): 293152 bytes

for 4001 users
2004/12/21  08:53:29  cybservd:649   Info   15458   Initialize service
(Service) (None)                Listening on TCP port 2345 for

incoming requests
2004/12/21  08:53:29  cybservd:472   Info   15458   Initialize service
(Service) (None)                CAS version 1.00 for HP-UX

started successfully (process

ID=15458)
2004/12/21  08:53:44  acpt_conn:376  Info   15459   New connection
(Session) pc1.company.com     Received connection from

pc1.company.com (CAS process

ID=15459)
2004/12/21  08:53:45  acpt_conn:376  Info   15461   New connection
(Session) pc1.company.com     Received connection from

pc1.company.com (CAS process

ID=15461)
2004/12/21  08:54:11  sighandlers:744 Info   15458   Session termination
(Service) (None)                Closed connection to

Pc1.company.com (CAS process

ID=15461)
```

```
2004/12/21  08:54:12  sighandlers:744 Info   15458    Session termination
(Service) (None)              Closed connection to

Pc1.company.com (CAS process

ID=15459)
2004/12/21  08:58:00  acpt_conn:376   Info   15483    New connection
(Session)  server1.company.com  Received connection from

server1.company.com (CAS

process ID=15483)
2004/12/21  08:58:00  msg_handlers:996Info   15483    Terminate Cyborg
Service(Session)  server1.company.com  The CAS service is being

terminated by session 15483
2004/12/21  08:58:00  sharmem:398     Info   15458    Service termination
(Service) (None)              Deallocated semaphore

(semaphore ID=299)
2004/12/21  08:58:00  sharmem:400     Info   15458    Service termination
(Service) (None)              Deallocated shared memory

(shared memory ID=4208)
2004/12/21  08:58:00  sighandlers:361 Info   15458    Service termination
(Service) (None)              CAS service shut down via CAS

Manager
```

☞   *See the Message format of the system log and trace files section for a description of the*
      message format.

Using a trace file
What is a trace file?
A trace file is a file that records the execution of CAS for tracing and debugging purposes.
This file is created only when tracing is enabled. Every CAS instance has its own, private
trace file, so the number of trace files can equal the number of CAS processes. Trace files
can contain binary data as part of their trace.

The following is an example of a trace file:

```
1998/12/21  08:53:44  acpt_conn:376  Info   15459    New connection
(Session)  pc1.company.com  Received connection from

pc1.company.com (CAS process

ID=15459)
1998/12/21  08:53:44  acpt_conn:379  Trace  15459    New connection
(Session)  pc1.company.com  New process for pc1.company.com

(pid=15459)
1998/12/21  08:53:44  sharmem:492     Trace  15459    Initialize service
(Session)  pc1.company.com  CAS process 15459 has been

registered
1998/12/21  08:53:44  proc_msg:106   Trace  15459    Getting new message
(Session)  pc1.company.com  Beginning to wait for messages

from the client
1998/12/21  08:53:44  read_msg:652   Trace  15459    Start Application request
(Session)  pc1.company.com  Start Application received
```

```
1998/12/21  08:53:44  read_msg:714   Trace  15459   Start Application request
(Session)  pc1.company.com   Start Application information
1998/12/21  08:53:44  read_msg:1305  Trace  15459   Start Application request
(Session)  pc1.company.com      Message version   = 0
1998/12/21  08:53:44  read_msg:1309  Trace  15459   Start Application request
(Session)  pc1.company.com      Application ID    = 3
1998/12/21  08:53:44  read_msg:1313  Trace  15459   Start Application request
(Session)  pc1.company.com      Use Cyborg account = Yes
1998/12/21  08:53:44  read_msg:1318  Trace  15459   Start Application request
(Session)  pc1.company.com      Environment name  = "env1"
1998/12/21  08:53:44  read_msg:1324  Trace  15459   Start Application request
(Session)  pc1.company.com      Username          = (not

applicable)
1998/12/21  08:53:44  read_msg:1339  Trace  15459   Start Application request
(Session)  pc1.company.com      Encrypt data      = Yes
1998/12/21  08:53:44  sharmem:1003   Trace  15459   Internal processing
(Session)  pc1.company.com  Updated information about CAS

process 15459
1998/12/21  08:54:12  start_app:846  Trace  15459   Application
cyborg     pc1.company.com  Waiting for data from either

client or host application
1998/12/21  08:54:12  start_app:972  Trace  15459   Output from host app
cyborg     pc1.company.com  Activity detected on stdout of

FILE01 Application
1998/12/21  08:54:12  start_app:994  Trace  15459   Output from host app
cyborg     pc1.company.com  Read 18 bytes from stdout of

FILE01 Application
1998/12/21  08:54:12  start_app:1001 Trace  15459   Output from host app
cyborg     pc1.company.com  Data from stdout:  "0007C14200

GOODBY"
1998/12/21  08:54:12  write_msg:348  Trace  15459   Output from host app
cyborg     pc1.company.com  Writing 18 bytes of data from

stdout to client
1998/12/21  08:54:12  write_msg:353  Trace  15459   Output from host app
cyborg     pc1.company.com  "0007C14200 GOODBY"
1998/12/21  08:54:12  write_msg:1009 Trace  15459   Start Application response
cyborg     pc1.company.com  Writing message header:

class=3, type=1, length=18
1998/12/21  08:54:12  start_app:1008 Trace  15459   Output from host app
cyborg     pc1.company.com  Successfully sent 18 bytes to

pc1.company.com
1998/12/21  08:54:12  start_app:846  Trace  15459   Application
cyborg     pc1.company.com  Waiting for data from either

client or host application
1998/12/21  08:54:12  start_app:846  Trace  15459   Application
cyborg     pc1.company.com  Waiting for data from either

client or host application
1998/12/21  08:54:12  start_app:972  Trace  15459   Output from host app
cyborg     pc1.company.com  Activity detected on stdout of

FILE01 Application
```

```
1998/12/21  08:54:12  start_app:989  Trace  15459    Output from host app
cyborg     pc1.company.com  stdout pipe has closed
1998/12/21  08:54:12  start_app:1349 Trace  15459    Session termination
cyborg     pc1.company.com  Checking if FILE01 Application

has terminated (attempt 1)
1998/12/21  08:54:12  start_app:1387 Trace  15459    Session termination
cyborg     pc1.company.com  FILE01 Application (process

ID=15460) exited normally
1998/12/21  08:54:12  start_app:623  Trace  15459    Session termination
cyborg     pc1.company.com  FILE01 Application has completed

successfully
1998/12/21  08:54:12  acpt_conn:405  Trace  15459    Session termination
cyborg     pc1.company.com  Communication with

pc1.company.com has completed

successfully (process ID=15459)
```

☞ *See the Message format of the system log and trace files section for a description of the message format.*

Filenames
Each instance of CAS creates its own trace file (located in the `/cyborghome/app/server/trace` directory). When there are many clients connected, the number of trace files increases similarly. The filenames of the trace files are designed to assist the administrator in finding the file for a particular client among the multitude of files.

▪ **For 'talking' CAS**
For each CAS process started by the execution of the client session ('talking' CAS), the filename is derived from the client hostname (or IP address if the hostname is not available) and process ID of the associated 'talking' CAS. The format of the filename is:

clientaddress_processID.**trc**

For example:
pc1.company.com_29159.trc

This indicates that the client address is `pc1.company.com` and that the process ID of the 'talking' CAS process is 29159.
The usage of process ID in the filename allows a single client to connect multiple times without erasing the previous log while clearly separating each connection.

Note: *If tracing is enabled and disabled several times within a single connection, all traces will be written to the same file, since the client and process ID of the 'talking' CAS remain the same.*

▪ **For 'listening' CAS**
'Listening' CAS was started by the CAS script. There is no connected client, so the filename is derived solely from the process ID of 'listening' CAS. The format of the filename is:

cybservd_*processID*.**trc**

For example:
cybservd_29150.trc

This indicates that 'listening' CAS has a process ID of 29150.

Trace file maintenance

It is the responsibility of the system administrator to remove or back up trace files. New files will be generated for each session when tracing is enabled.

Message format of the system log and trace files

The following diagram illustrates the fields found in each message:
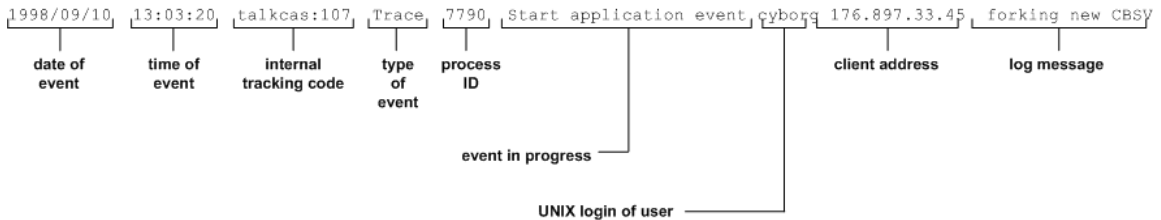
Each message (record) will contain the following nine fields separated by tabs:

Name	Description
Date of event	Date of the event. The date format is YYYY/MM/DD.
Time of event	Time of the event. The time format is HH:MI:SS. The time is taken from the local system time on the UNIX server.
Internal tracking code	For internal use only.
Type of event	The type of event. Possible values are: Error (error) Info (informational) Trace (trace)
Process ID	The process ID of the CAS process that is logging the message.
Event in progress	The event in progress when the log entry was created. Examples are: New connection Getting new message
UNIX login of user	The identity under which the server application is running. For the Cyborg user, this will be 'cyborg'.
Client address	The hostname of the GUI client connected to the 'talking' CAS which is logging the event. If the hostname is not available, then the IP address is printed in dotted decimal notation.
Log message	The actual log message.

-killtalking

Terminate all client application programs

Use this switch to terminate all application programs that are currently running.

casmgr **-killtalking**

Even after this command is used, CAS can still start new application programs if it is enabled. The 'disable CAS' switch (-disable) is often used before this command to prevent new server application programs from starting.

Use the 'obtain a list of connected sessions' switch (-sessions) to verify that the application programs were terminated.

Note: *This command requires a password.*

-notraceall

Disable the tracing of every CAS process

Use this switch to disable tracing of CAS for all sessions:

casmgr **-notraceall**

This will disable the tracing of the CAS process and general I/O through CAS for every session: sessions with host applications, sessions managing administrative sessions, and the listening CAS session.

Tracing can be enabled using the 'enable tracing of every CAS process' switch (-traceall) or by using the 'enable tracing for a single application program for a single client' switch (-tracesession).

Note: *This command requires a password.*

-notracedefault

Disable the tracing of CAS for subsequent clients

Use this switch to disable tracing for all new application programs:

casmgr **-notracedefault**

This will cause subsequent application programs to not trace immediately upon their startup. Currently executing application programs are not affected.

Tracing can be enabled using the 'enable tracing of every CAS process' switch (-traceall) or by using the 'enable tracing for a single application program for a single client' switch (-tracesession).

Note: *This command requires a password.*

-notracesession

Disable tracing of CAS for a single host application for a given client

Use this switch to disable tracing of CAS for a single application program for a single client:

casmgr **-notracesession:**_sessionID_

This will disable the tracing of the CAS process and I/O through CAS for single application program for a particular GUI client. The application program is identified with a session ID, as determined by the 'obtain a list of connected sessions' switch (`-sessions`).

Note: *This command requires a password.*

-sessions
Obtain a list of connected sessions

Use this switch to obtain a list of connected sessions:

casmgr -**sessions**

 This will print the list of sessions connected to CAS with the following information:

- Application ID of the server application
- Environment of the server application
- Login of the user using the server application
- Starting date and time of the application program
- Process ID of the CAS process on the server that is responsible for the server application
- Hostname or IP address of the client
- Whether tracing is enabled for this application

-traceall
Enable tracing of every CAS process

Use this switch to enable tracing of CAS for all CAS processes:

casmgr **-traceall**

This will enable the tracing of the CAS process and general I/O through CAS for every session: sessions with host applications, sessions managing administrative sessions, and the primary server session.

Note: *This command requires a password.*

-tracedefault
Enable tracing of CAS for subsequent clients

Use this switch to enable tracing for all new application programs:

casmgr **-tracedefault**

This will cause subsequent application programs to begin tracing immediately upon their startup. Currently executing sessions are not affected.

Tracing can be disbled using the 'disable tracing of every CAS process' switch (`-notraceall`) or by using the 'disable tracing for a single application program for a single client' switch (`-notracesession`).

Note: *This command requires a password.*

-tracesession
Enable tracing for a single application program for a single client

Use this switch to enable tracing of CAS for a single server application program for a specific client:

casmgr **-tracesession:***sessionID*

This will enable the tracing of the CAS process and I/O through CAS for a single server application program for a particular GUI client. The application program is identified with a session ID, as determined by the 'obtain a list of connected sessions' switch (`-sessions`).

Note: *This command requires a password.*

■ *Using a trace file*
For more information about trace output

-version
Determine the version of CAS

Use this switch to determine the version of CAS:

casmgr **-version**

This will print the version of CAS Manager and CAS. It will also print the operating system on which CAS is running.

Viewing the system log and trace files

Although the system log and trace files are basically text files and can be viewed in any editor, the long text lines are difficult to manage. The scripts `viewlog` and `viewlogmsg` in the `/cyborghome/app/server` directory can be used to view the system log and trace files.

Both `viewlog` and `viewlogmsg` are awk scripts that use the standard **awk(1)** utility. Both scripts take the filename of the log or the trace file as their argument. If no argument is given, then they will read from standard input.

The script `viewlog` prints all the details from its input file in a more readable format, spreading the information across multiple lines and labeling each field. The script `viewlogmsg` prints only the log message without any of the other fields.

Examples:
```
viewlog system.log | more

viewlogmsg trace/pc1.company.com_29159.trc | more

tail -f system.log | viewlogmsg
```

Troubleshooting the CAS daemon (UNIX)

CAS installation error messages
If the expected output is not displayed when starting CAS, then check the following error message table for a possible solution and then re-execute the `cas` script.

Shell	Error	Problem	Solution
bourne korn c	cas: execute permission denied ksh: cas: cannot execute cas: Permission denied.	The cas script does not have executable permissions	At the operating system command prompt, enter: chmod u+x cas
bourne korn c	cybservd: execute permission denied ksh: cybservd: cannot execute cybservd: Permission denied.	The cybservd binary does not have executable permissions	At the operating system command prompt, enter: chmod u+x cybservd
bourne korn c	cas: not found ksh: cas: not found cas: Command not found.	The cas script is not in the: /cyborghome/app/ server directory	Change to the /cyborghome/app/server directory, and verify that the cas script exists by using ls cas.
bourne korn c	cybservd: not found ksh: cybservd: not found cybservd: Command not found.	The cybservd binary is not in the /cyborghome/ app/server directory	Change to the /cyborghome/app/server directory, and verify that cybservd exists by using ls cybservd.
bourne korn c	cybservd: Port 9888 is already in use by another process Cyborg Application Server version 1.04 terminated.	Other pre-existing software might already be configured to use this port and could be conflicting with CAS	You must change the port used by CAS by editing the /etc/services file. The administrator is responsible to ensure that each GUI client knows that CAS is no longer at 9888 but rather some other port.
		The CAS daemon is already running	Verify that the CAS daemon is running by using the following command: casmgr -isrunning

Shell	Error	Problem	Solution
bourne korn c	If you use the following command: ps -ef | grep cybservd and no processes are found	The cybservd process has aborted	Enable tracing of the CAS session immediately upon startup, so that all system error and trace messages will be sent to the trace file. To enable tracing at startup, edit the cas script. Add '-t' to the last command in the file. You must then log in as root and execute the script cas. After the process again aborts, examine the trace file using viewlogmsg to determine the reason for the process abort.
bourne korn c	cyborg: No such user on local machine	The Cyborg user has not been created	You must create the Cyborg user 'cyborg' in the operating system.

Client installation error messages

If the following error occurs:
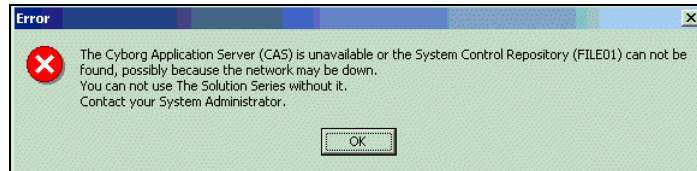
Try one of the following:

Problem	Solution
The server is not a valid machine.	Correct the 'Host' field in the Connection Editor for the environment in question. The server will differ from installation to installation.
The server exists but the port is not a valid number.	Correct the 'Port' field in Connection Editor for the environment in question. The port number is listed in the /etc/services file on the server under the service name of 'cyborg'. The recommended value is 9888.
The server exists but is not the correct machine.	Use the Connection Editor to examine the connection properties and ensure that the listed server host is the machine on which CAS is running.

Problem	Solution
The server is correct but the port is incorrect.	Use Connection Editor to examine the connection properties and ensure that the port is correct. The port is usually 9888.
CAS is not started.	Make sure that CAS is running on the server. To launch CAS, log in as root and execute the script cas.

CAS Manager messages

Following is a list of messages you may encounter from the CAS Manager.

Additional error messages: command line parsing

Message	Condition
casmgr: Invalid switch invalidswitch	Invalid switch on command line
casmgr: Missing colon for invalidswitch	Missing colon for any switch that requires an argument
casmgr: Value required after invalidswitch	No argument after any switch except -password that requires an argument (see note below table for -password)
casmgr: Bad number invalidnumber	A badly-formed number is given where a number is expected
casmgr: Too many passwords specified	Two or more passwords on command line
casmgr: Too many commands specified	Two or more commands on command line
(Display the syntax for the command)	No arguments given on command line
casmgr: No command specified	No command on command line, but a password was specified

Note   *It is not an error to omit the password after the* `-password` *switch. This situation corresponds to using the empty string as the password.*

Additional error messages: network I/O

Message	Condition
casmgr: Insufficient memory	Memory allocation failure
casmgr: CAS is not running on server, port nnnn.	CAS is not running on the current UNIX host at the TCP port specified in the /etc/services file under 'cyborg'
casmgr: Unable to send data to server	Write error while writing to socket

Message	Condition
casmgr: Unable to receive data from server	Read error while reading from socket

Warning messages

Warning messages are printed by CAS Manager or standard output,but do not prevent the command from completing.

Message	Condition
casmgr: Warning: password not needed	Single password on command line, but a password is not needed by the command

Stopping CAS

In the event of a payroll run or backup
All environments

To prevent online usage to all environments, we suggest that you perform the following steps:

1. casmgr -disable
   This prevents new users from signing on.
2. casmgr -killtalking
   This terminates any remaining processes, if required.
3. casmgr -enable
   Once batch processing or backup is concluded, this will allow online usage.

A single environment

To prevent online usage to one environment, we suggest that you perform the following steps:

1. Edit the cybenv.cfg file, locate the record associated with application ID 3 for this environment and change the field labeled 'enabled' to 'N'.
   This prevents new users from signing on to this one environment
2. casmgr -killtalking
   This terminates any remaining processes, if required.
3. Once batch processing or backup is concluded, edit the cybenv.cfg file, locate the record associated with application ID 3 for this environment and change the field labeled 'enabled' to 'Y'.

Uninstalling CAS

To uninstall the CAS daemon, perform the following steps:

1. Stop the CAS daemon (casmgr -killdaemon).
2. Delete the CAS files and directories:
   cd *cyborghome*/app/server
   rm cybservd cas cybapp.cfg cybenv.cfg viewlog viewlogmsg
   rm -r system.log trace
3. Edit the file /etc/services to remove the Cyborg port to the network services database.

The entries in the file are usually given in numeric order, so find the location where 9888 should appear, then remove the following line of the file:

cyborg   9888/tcp   # Assigned by IANA to Cyborg Systems

4. Save the file and exit the editor.

A P P E N D I X   F

Configuring Kernel Parameters on UNIX

In This Appendix

Configure Kernel Parameters

UNIX resources facilitate inter-process communication. Review the table below for the recommended platform settings. If your parameter settings are insufficient, you might experience operational errors after installation. Remember to make a backup copy of your UNIX kernel. Consult your operating system documentation for additional information. For Oracle, consult the Oracle Installation documentation for additional requirements. Recommended Kernel Parameters per Operating System:

- For each Solution Series online session, processes = 2 (CYBIO + CBSVO)
- For Interactive Workforce, processes = (max # of CYBIO) + (max # of CBSVO) connections
- For each Web Client session, processes = 2 (CYBIO + CBSVO)

Operating System	Maximum number of ...	Kernel Parameter	Recommended
AIX RS6000	Open files per system	No parameter defined	NA
	Processes per User	maxuproc	**((nproc*9)/10)
	Processes per system	No parameter defined	NA
	Processes per system using semaphore "undo" structures	No parameter defined	NA
	Max # of file locks per system	No parameter defined	**nproc * 12
HP-UX	Open files per system	nfile	(15*nproc + 2048)
	Processes per User	Maxuprc	((nproc*9)/10)
	Processes per system	nproc	(20 + 8 * maxusers)
	Processes per system using semaphore "undo" structures	semmnu	nproc + 4
	Max # of file locks per system	No parameter defined	nproc * 12
Sun Solaris	Open files per system	No parameter defined	NA
	Processes per User	maxuprc	((nproc*9)/10)

Operating System	Maximum number of ...	Kernel Parameter	Recommended
	Processes per system	max_nprocs	(20 + 8 * maxusers)
	Processes per system using semaphore "undo" structures	seminfo_semmnu	nproc + 4
	Max # of file locks per system	No parameter defined	nproc * 12

**AIX should not require kernel parameter configuration as it will adjust itself to handle any number of processes; however, certain system parameters may need to be adjusted to increase performance.

Hewitt

The Application and Reporting Solution Server:

Requirement	Description
Operating System	HP-UX 11i, HP 9000
RAM	1GB
Disk Space	• 1GB in addition to operating system requirements for up to 2000 employees[1] • Allow an additional 100MB for each additional 1000 employees • Allow additional 500MB space for temporary files
Processor Speed	650 MHz
Media Access	Access to CD-ROM drive (either locally or on a network)
Software	• Micro Focus Server Express 4.0 SP2 (32 bit) • Oracle 9i Database Server / ProCobol (relational only) • NFS or other products such as SAMBA • C Compiler HP (B3901BA b.11.11.04) • HP a C++ runtime library patches PHSS 24638 • Java Runtime Environment (JRE) 1.4.2_06

The Administrative Client:

Requirement	Description
Operating System	Microsoft Windows 2000 and Window XP Professional
RAM	256MB
Disk Space	250MB
Processor Speed	500MHz
Media Access	Access to a CD-ROM drive (either locally or on a network)
Software	Microsoft Internet Explorer 6.0
License	• Micro Focus Runtime license • Oracle Runtime License (for Oracle DB Server)

---

[1] Assumes 2000 employees (1700 weekly and 300 semi-monthly) in five organizations retaining four months history and labor in a relational environment.

Hewitt

The Administrative Client (continued):

Email Events	• SMTP • MAPI (like Outlook) • Lotus Notes: **Running Windows XP** – Lotus Notes 6.x (Notes 6 series) required plus Outlook XP and Domino Access for Microsoft Outlook (DAMO) corresponding to the Notes version. **Running Windows 2000** – Lotus Notes 5.0.10 or higher (Notes 5 series) plus Outlook 2000.
Letter Events	Microsoft Word
Import Functionality	Microsoft Excel